

PULSEIRA INTELIGENTE PARA MONITORIZAÇÃO DE FUNÇÕES VITAIS

Pedro Manuel Araújo Santos Godinho



Mestrado em Engenharia Eletrotécnica e de Computadores

Área de Especialização de Sistemas Autónomos

Departamento de Engenharia Eletrotécnica

Instituto Superior de Engenharia do Porto

Este relatório satisfaz, parcialmente, os requisitos que constam da Ficha de Disciplina de Tese/Dissertação, do 2º ano, do Mestrado em Engenharia Eletrotécnica e de Computadores

Candidato: Pedro Manuel Araújo Santos Godinho, N° 1060275, 1060275@isep.ipp.pt

Orientação científica: Engº Eduardo Silva, eaps@lsa.isep.ipp.pt



Mestrado em Engenharia Eletrotécnica e de Computadores

Área de Especialização de Sistemas Autónomos

Departamento de Engenharia Eletrotécnica

Instituto Superior de Engenharia do Porto

13 de novembro de 2013

Agradecimentos

Gostaria de começar por agradecer aos meus pais por todo o apoio, confiança e, por vezes, sacrifícios que fizeram para me proporcionar melhores condições não só neste mas em vários aspetos da minha vida. Sem eles não seria possível a conclusão desta etapa. Um sincero obrigado a eles!

A minha irmã pela disponibilidade que sempre teve para comigo.

Ao meu orientador, Eng^o Eduardo Silva pela boa disposição mantida, o apoio, compreensão e pela confiança demonstrada no meu trabalho desde o seu início.

Ao Carlos Silva, Karla Gonçalves e a todos que sempre se mostraram disponíveis para ajudar a ultrapassar as dificuldades.

Por fim, mas não menos importante, também a todos aqueles que tão importantes para mim foram ao longo de todo o curso e na minha vida.

Um sentido obrigado a todos,

Pedro Araújo Godinho

Resumo

A crescente utilização de dispositivos móveis com diferentes finalidades é uma realidade. Com estes dispositivos, o utilizador tem a necessidade de aceder e usar dados em tempo real provenientes de diversas fontes. Uma tendência acentuada passa pela incorporação destes dispositivos móveis no vestuário, designados por dispositivos *wearable*.

Segundo a empresa IMS Research, o mercado deste tipo de dispositivos irá aumentar de 14 milhões de unidades registadas no presente ano (2013), para cerca de 171 milhões em 2016, sendo esta previsão conservadora, segundo o analista da IMS Research, Theo Ahadome [15]. A maioria dos dispositivos portáteis está atualmente projetada para questões de saúde, como a monitorização do nível de glicose e batimento cardíaco.

O objetivo deste trabalho passa por definir e implementar um dispositivo *wearable* para aplicações de saúde com um conjunto de funcionalidades para monitorização dos sinais vitais do utilizador. Posteriormente esta base pode ser aplicada em cenários de aplicação distintos, em que todos os dispositivos comunicam entre si, e fazem o reencaminhamento da informação para onde mais interessar. Foi desenhado e implementado hardware e software, para a construção de aplicações capazes de realizar a monitorização do batimento cardíaco, temperatura e humidade corporal, deteção de quedas, qualidade do sono, e chamadas de emergência.

Este trabalho aborda os diferentes cenários e aplicações da utilização deste dispositivo, invocando as necessidades específicas de cada situação, sendo estas necessidades trabalhadas e transformadas em características e especificações do sistema.

A plataforma de hardware e software permite criar um ecossistema de aplicações, permitindo usar todas as infraestruturas do sistema desenvolvido em futuros cenários de aplicação.

Palavras-Chave

Wearable, monitorização, atividade física, sinais vitais, dispositivos móveis, sistemas tempo real, hardware baixo consumo.

Abstract

The growing use of mobile devices with different goals is a reality.

Based on this devices, the user has a need to have access and to use real time data provided from several sources. The inclusion of this mobile devices on the clothing is a major trend, designated by wearable devices

IMS Research reported that the market of this type of devices will grow from 14 million registered units by the present year (2013), to about 171 million in 2016, being a very conservative estimation, referred by the IMS Research analyst Theo Ahadome [15]. The major part of the mobile devices are actually designed for health applications, as glucose monitoring and heart rate.

The goal of this work is to define and implement a wearable device for health application with a set of facilities for vital signals monitoring of the user. Later this base can be applied in different application scenarios, in which all devices communicate with each other, and make the forwarding of information to where is more interesting. It was designed and implemented hardware and software for applications capable to measure the heart rate, body temperature and humidity, fall detection, activity and sleep monitoring and emergency calls.

This work approaches the different scenarios and applications of the use of this device, describing the specific needs of each situation and working and transforming into system characteristic and specifications.

The hardware and software platform enables to create an applications ecosystem with granted access to the overall infrastructures of the developed system, in future application scenarios.

Keywords

Wearable, monitoring, physical activity, vital signs, mobile devices, real time systems, low power hardware.

Índice

AGRADECIMENTOS.....	I
RESUMO	IV
ABSTRACT.....	VII
ÍNDICE	X
ÍNDICE DE FIGURAS.....	XIII
ACRÓNIMOS	XVI
1. INTRODUÇÃO	1
1.1. O QUE É UM DISPOSITIVO <i>WEARABLE</i>	1
1.2. BREVE HISTÓRIA DOS DISPOSITIVOS <i>WEARABLE</i>	3
1.3. MOTIVAÇÃO E ENQUADRAMENTO.....	4
1.4. OBJETIVOS	5
1.5. ORGANIZAÇÃO DO RELATÓRIO	6
2. CONCEITOS FUNDAMENTAIS	7
2.1. DISPOSITIVOS <i>WEARABLE</i>	7
2.1.1. MODOS DE OPERAÇÃO.....	7
2.1.2. ATRIBUTOS DE UM DISPOSITIVO <i>WEARABLE</i>	9
2.2. CUIDADOS DE SAÚDE E DE VIDA ASSISTIDA	9
2.2.1. BATIMENTO CARDÍACO	9
2.2.2. TEMPERATURA / HUMIDADE CORPORAL.....	10
2.2.3. DETECÇÃO DE QUEDAS	10
3. ESTADO DA ARTE.....	13
3.1. CONCEITOS.....	13
3.1.1. ASSISTENTE INTERATIVO	13
3.1.2. REALIDADE AUMENTADA	14
3.1.3. RECONHECIMENTO DO CONTEXTO.....	14
3.2. APLICAÇÕES.....	15
3.2.1. FORMAÇÃO E SUPORTE	15
3.2.2. APLICAÇÕES DE ASSISTÊNCIA PARA DESABILITADOS	15
3.3. EQUIPAMENTOS / PROTÓTIPOS	16
3.3.1. MYO (GESTURE ARMBAND CONTROLLER)	16
3.3.2. BATIMENTO CARDÍACO ATRAVÉS DO SOM.....	17
3.3.4. SONY SMARTWATCH 2	19
3.3.5. JAWBONE UP	20
3.3.6. MISFIT SHINE	21

4. TECNOLOGIAS E FERRAMENTAS DE DESENVOLVIMENTO.....	23
4.1. EM::BLOCKS (C/C++ IDE)	23
4.2. RTX - SISTEMA OPERATIVO	24
4.3. ALTUM DESIGNER	25
4.4. SOLIDWORKS	28
5. ESPECIFICAÇÃO E DESCRIÇÃO DO SISTEMA.....	31
5.1. VISÃO GERAL DO SISTEMA.....	31
5.2. REQUISITOS GENÉRICOS DO SISTEMA	32
5.3. CENÁRIOS DE APLICAÇÃO	35
6. ARQUITETURA DE HARDWARE.....	43
6.1. MÉTODOS DE ENTRADA.....	45
6.2. MÉTODOS DE SAÍDA	45
6.3. COMUNICAÇÕES	45
6.4. FONTES DE ENERGIA.....	46
6.5. DESCRIÇÃO DOS COMPONENTES	48
6.5.1. MÚLTIPLA CONECTIVIDADE	48
6.5.2. CONECTIVIDADE BLUETOOTH	49
6.5.3. ACELERÓMETRO.....	50
6.5.4. MICROCONTROLADOR	50
6.5.5. ECRÃ.....	51
6.5.6. BATERIA.....	52
6.5.7. SENSOR DE TEMPERATURA.....	53
6.5.8. RESISTÊNCIA DA PELE.....	53
6.5.9. BATIMENTO CARDÍACO.....	54
7. ARQUITETURA DE SOFTWARE.....	55
7.1. ARQUITETURA GERAL	55
7.2. DEVICE DRIVERS	56
7.3. SISTEMA OPERATIVO	68
7.4. MIDDLEWARE.....	75
7.5. APLICAÇÕES	79
7.5.1. BATIMENTO CARDÍACO.....	79
7.5.2. TEMPERATURA E HUMIDADE CORPORAL.....	81
7.5.3. DETECÇÃO DE QUEDAS	82
7.5.4. MONITORIZAÇÃO DE SONO	85
7.5.5. INFORMAÇÃO DE IDENTIFICAÇÃO.....	87
7.5.6. CHAMADA DE EMERGÊNCIA	89
8. ARQUITETURA MECÂNICA.....	91
8.1. DISPOSIÇÃO DA ELETRÓNICA	91
8.2. <i>DESIGN</i> PROTÓTIPO.....	93
9. CONCLUSÕES.....	95

REFERÊNCIAS DOCUMENTAIS.....	97
ANEXO A. DIMENSÕES EXTERIORES DO ECRÃ	99

Índice de Figuras

Figura 1	Evolução das invenções criadas por Steve Mann's.....	3
Figura 2	Cenário de aplicação usando Sistemas Autónomos	4
Figura 3	Tipo de planos: Sagital (a), Coronal (b)	11
Figura 4	Tipo de quedas: Consciente (a), Inconsciente (b)	11
Figura 5	MYO	16
Figura 6	Batimento Cardíaco Através do Som	17
Figura 7	GIST	18
Figura 8	Sony SmartWatch 2.....	20
Figura 9	Aspeto do Jaybone UP	20
Figura 10	Aspeto do Misfit Shine.....	21
Figura 11	Misfit Shine a indicar a atividade diária.....	21
Figura 12	Posição do Misfit Shine para sincronizar	22
Figura 13	Aspeto do ambiente de desenvolvimento Em::Blocks	24
Figura 14	Arquitetura do RTOS RTX	25
Figura 15	Esquema elétrico	26
Figura 16	Modelo 3D do microcontrolador.....	26
Figura 17	Desenho preliminar	28
Figura 18	Ferramenta Sketch.....	29
Figura 19	Ferramenta Extrude	29
Figura 20	Ferramenta CutExtrude	30
Figura 21	Aparência do equipamento desenhado	30
Figura 22	Interação do dispositivo com o mundo.....	32
Figura 23	Aplicação do batimento cardíaco	36
Figura 24	Aplicação da temperatura.....	37
Figura 25	Aplicação de deteção de quedas	38
Figura 26	Aplicação de monitorização do Sono	39
Figura 27	Aplicação da chapa de identificação	41
Figura 28	Funcionamento básico do sistema.....	43
Figura 29	Arquitetura geral do hardware.....	44
Figura 30	Zonas de comunicação em aplicações <i>wearable</i>	46
Figura 31	Diferentes fontes de energia	47
Figura 32	Diagrama de blocos do Broadcom BCM43341.....	49
Figura 33	Modelo do módulo de Bluetooth PAN1323.....	49
Figura 34	a) Aspeto da caixa do integrado b) Esquema do <i>footprint</i> do integrado	50

Figura 35	Diagrama de Blocos do microcontrolador	51
Figura 36	Aspeto do ecrã de 1.28"	52
Figura 37	Estados de carga de uma bateria de polímeros de lítio	52
Figura 38	Sensor de temperatura	53
Figura 39	Pinos para medir a resistência da pele	54
Figura 40	Aspeto do sensor ótico	54
Figura 41	Arquitetura geral de <i>software</i>	56
Figura 42	Estrutura de comando para apagar o ecrã	57
Figura 43	Estrutura de comando para escrever uma linha	58
Figura 44	Fluxograma do driver de leitura do TCRT1010	60
Figura 45	Fluxograma do driver de leitura da resistência da pele	61
Figura 46	Fluxograma do driver do sensor TMP006	62
Figura 47	Comportamento do sensor capacitivo	63
Figura 48	Fluxograma da rotina de interrupção do TOUCH	65
Figura 49	Exemplo da máquina de estados para a função acordar	66
Figura 50	Fluxograma da comunicação com o MEMS	67
Figura 51	Estrutura do CMSIS-RTOS RTX	68
Figura 52	Estados das <i>threads</i>	69
Figura 53	Sequência de operação de um <i>Mutex</i>	70
Figura 54	Operações dos semáforos	71
Figura 55	Estrutura de uma <i>Message Queue</i>	72
Figura 56	Estrutura de um <i>Mail Queue</i>	72
Figura 57	Sequência temporal de uma função <i>timer</i>	73
Figura 58	Gestor de Recursos (<i>Resources Management</i>)	75
Figura 59	Gestor da Interface de utilizador (<i>User Interface Management</i>)	76
Figura 60	Gestor de Eventos (<i>Event Management</i>)	77
Figura 61	Menu de Batimento Cardíaco	79
Figura 62	Implementação real da aplicação de batimento cardíaco	79
Figura 63	Fluxograma para aceder à aplicação do batimento cardíaco	80
Figura 64	Menu de Temperatura e Humidade	81
Figura 65	Implementação real da aplicação de Temperatura e Humidade	81
Figura 66	Fluxograma para aceder à aplicação de temperatura e humidade	82
Figura 67	Menu de Detecção de Quedas	83
Figura 68	Ecrã após deteção de queda	83
Figura 69	Fluxograma para aceder à aplicação de temperatura e humidade	84
Figura 70	Menu de Monitorização do Sono	85
Figura 71	Menu Principal para entrar na aplicação de Monitorização do Sono	85
Figura 72	Fluxograma da aplicação de monitorização do sono	86
Figura 73	Menu de Informação de Identificação	87

Figura 74	Aplicação em modo de transmissão de informação	87
Figura 75	Fluxograma da aplicação de Identificação	88
Figura 76	Menu de indicação de chamada de emergência em curso	89
Figura 77	Indicação de chamada de emergência em curso no ecrã real	89
Figura 78	Ecrã Selecionado	92
Figura 79	<i>Design</i> provisório do botão de emergência	92
Figura 80	<i>Design</i> protótipo, parte frontal (a) e parte posterior (b).	93
Figura 81	Simulação do aspeto do dispositivo quando em utilização	93

Acrónimos

API	- Application Programming Interface
APP	- Application
BGA	- Ball Grid Array
BT	- Bluetooth
ECG	- Eletrocardiograma
EDR	- Enhanced Data Rate
GPS	- Global Positioning System
I2C	- Inter Integrated Circuit
IEEE	- Institute of Electrical and Electronics Engineers
KB	- KiloByte
LCD	- Liquid Crystal Display
LED	- Light Emitting Diode
LGA	- Land Grid Array
MEMS	- Micro Electro-Mechanical Systems
MHz	- Megahertz
NFC	- Near Field Communication
RAM	- Random-Access Memory
RC	- Resistance-Capacitance
RF	- Radio-Frequency
RFID	- Radio-Frequency IDentification
ROM	- Read-Only Memory
RTC	- Real-Time Clock
RTOS	- Real-Time Operating System
RTX	- Real-Time eXecutive
SOS	- Save our Souls
SPI	- Serial Peripheral Interface
USB	- Universal Serial Bus
WLBGA	- Embedded Wafer Level Ball Grid Array

1. INTRODUÇÃO

1.1. O QUE É UM DISPOSITIVO *WEARABLE*

Uma das primeiras noções de um computador *wearable* foi mencionado num relatório técnico de autoria de Thad Starner em 1995 chamado "*The Cyborgs are coming*". Neste documento o autor sugeriu que a *interface* de um equipamento *wearable* tem duas características, persistência e constância. Persistência descreve que a interface do *wearable computer* é persistente, uma vez que está sempre disponível mesmo sendo usada simultaneamente com outras tarefas. Constância descreve a propriedade em que a *interface* do *wearable computer* é usada em cada situação. Obviamente, esta definição é mais direcionada para um *wearable computer* que esteja totalmente integrado no dia-a-dia do seu utilizador.

O termo "*Cyborg*" foi originalmente cunhado por Manfred Clynes e Nathan Kline em 1960. Eles descreveram um ciborgue como uma combinação de um ser humano e uma máquina na qual a interface se tornou uma extensão natural, que não requer a atenção contínua para ser controlada. Embora Clynes e Kline tenham usado astronautas no seu trabalho, ele também pode ser usado em *wearable computers* como uma metáfora para expressar uma integração perfeita entre a interface e o seu utilizador [2].

Em 1997, Rhodes definiu as características de um *wearable computer*, tendo este cinco propriedades:

- É portátil quando está a ser usado
- Possibilita ter as mãos livres

- Pode chamar a atenção do utilizador mesmo quando este não o esteja a usar
- Está sempre ligado
- Tenta perceber em que contexto está o utilizador, de modo a poder servi-lo melhor

Uma das principais características que diferencia um dispositivo *wearable* de um simples computador, ou equipamento informático portátil, é a possibilidade deste dispositivo permanecer anexado ao corpo do utilizador durante um longo período de tempo sem que se torne incómodo ou condicionador de movimentos.

Por outras palavras um dispositivo *wearable* é por norma um dispositivo pequeno, portátil e interativo que está sempre pronto para ser utilizado. Este encontra-se, geralmente integrado em objetos do quotidiano como: as roupas, pulseiras, relógios, óculos entre outros. Para a definição de um dispositivo *wearable* pode recorrer-se aos principais modos de operação e atributos.

1.2. BREVE HISTÓRIA DOS DISPOSITIVOS *WEARABLE*

Algumas inovações têm ajudado a definir um caminho para os *wearable computers*. Podemos ir até às datas de 1268 quando se ouviu falar dos óculos pela primeira vez e a invenção do relógio de bolso em 1762. O primeiro *wearable computer* foi inventado pelo Ed Thor e Claude Shannon em 1966 com a intenção ser usado por um assistente para definir a velocidade de uma roleta. Nesse mesmo ano foi criado por Sutherland, o primeiro ecrã baseado num computador, para ser colocado na cabeça.

Os *wearable computers* têm percorrido um longo caminho ao longo destes anos. No ano de 1980 o laboratório de *wearable computers* da Universidade de Stanford tentou criar o software necessário para colocar um servidor web num computador do tamanho de uma caixa de fósforos. Este tipo de tecnologia tem contribuído para a possibilidade de produção de *wearable computers* úteis e discretos fazendo com que as empresas mostrem algum interesse neste tipo de tecnologia [1].

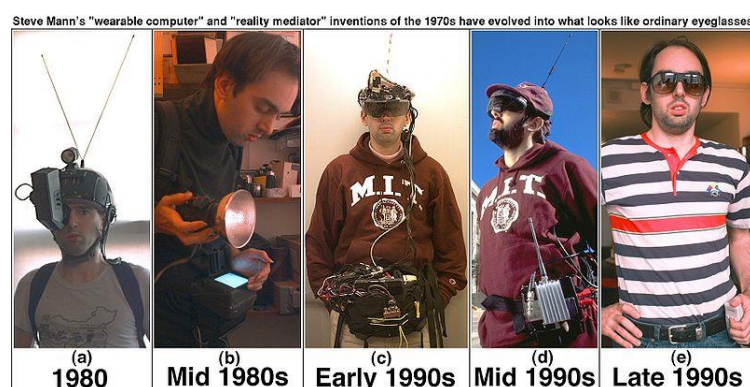


Figura 1 Evolução das invenções criadas por Steve Mann's

Entre os anos 1991 e 1993, três *wearable computers* foram construídos como demonstradores de tecnologia pela Universidade de Carnegie Mellon. Estes dispositivos, denominados de VuMan1, VuMan2 e Naigator1, foram construídos como projetos de aulas e não tinham nenhum utilizador definido. Todos eles usaram o Private Eye como o dispositivo de visualização. O Private Eye é um dispositivo com resolução *Color Graphics Adapter* (CGA) e que é apropriado para desenhos de baixa resolução.

1.3. MOTIVAÇÃO E ENQUADRAMENTO

O LSA/ISEP tem vindo a desenvolver aplicações robóticas em diversas vertentes, em que uma delas é a busca e salvamento. Neste caso concreto, nem sempre é possível ao sistema autónomo concluir a tarefa sem o auxílio de meios humanos, que nesta aplicação poderão estar sujeitos a condições perigosas e adversas, e em locais de difícil acesso ou remotos. Como tal a necessidade de monitorizar os sinais vitais da equipa ou mesmo das pessoas a serem salvas é fundamental, mas estes de nada servem se não forem partilhados com quem de interesse. Uma possível solução para este problema, passaria por usar as capacidades de comunicações do sistema autónomo para redirecionar a informação recolhida, conforme exemplo da Figura 2. Esta poderia ter sinais vitais, posição GPS ou outra forma de localização, e identificar a pessoa.

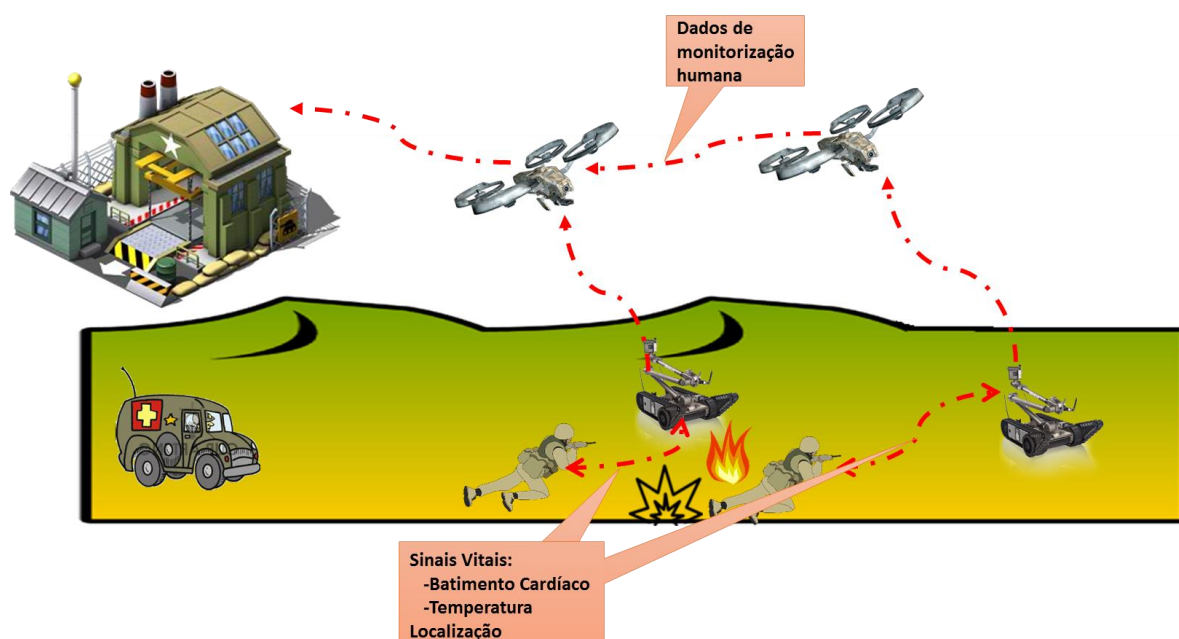


Figura 2 Cenário de aplicação usando Sistemas Autónomos

Outro exemplo seria o caso de um utilizador com problemas cardíacos, que recorre frequentemente a uma farmácia para monitorizar as condições de saúde. Apesar de esta ser uma rotina bem presente no nosso dia-a-dia, não se mostra vantajosa comparada, por exemplo, com a possibilidade do utilizador passar a utilizar um equipamento *wearable* no seu pulso. Deste modo, a sua condição seria monitorizada constantemente de uma forma elegante e prática, e no caso de algum valor lido estar fora do esperado despoletaria alertas.

Como engenheiros de desenvolvimento na área de sistemas embebidos, o interesse em acompanhar as novas tendências tecnológicas é substancial, e o fato de poder criar uma aplicação que possa ser uma mais-valia em situações críticas é por si só motivador. Aliado a um especial interesse nas aplicações médicas e de proteção da vida humana, que se evidenciou na licenciatura, cujo tema do trabalho de estágio foi “Alerta de Imersão e Afastamento”, que consistia em alertar um responsável sobre um possível acontecimento de afogamento, ou excessiva distanciação da pessoa a ser monitorizada, recorrendo ao protocolo de comunicação ZigBee. O desenvolvimento desta plataforma, e a sua posterior incorporação em uma aplicação de busca e salvamento, ou apenas na simples monitorização de sinais vitais num formato *wearable*, é um desafio que poderá ser aproveitado para a criação de outras futuras aplicações.

1.4. OBJETIVOS

Com este trabalho pretende-se estudar, especificar, projetar e implementar um dispositivo/plataforma com o intuito de ser uma base para futuros dispositivos *wearable*, com mais ênfase nas áreas da saúde.

O projeto centraliza-se na criação genérica da plataforma de hardware e software bem como todas as especificidades associadas aos cenários explorados.

Sendo assim, os principais objetivos a atingir são:

- Estudo e identificação dos requisitos e dos conceitos fundamentais aplicáveis
- Estudo e especificação da plataforma de hardware:
 - Plataforma base
 - Hardware específico para aplicações de saúde
- Estudo e especificação da *framework* de software
- Desenho e implementação dos cenários de aplicação:
 - Batimento Cardíaco
 - Oxímetro
 - Temperatura e Humidade Corporal
 - Detecção de Quedas
 - Monitorização de Sono
 - Chapa de Identificação
 - Chamada de Emergência
- Estudo e desenho da arquitetura mecânica tendo em vista os cenários de aplicação

1.5. ORGANIZAÇÃO DO RELATÓRIO

No capítulo Introdução é feita uma apresentação e contextualização do tema abordado neste trabalho, com uma apresentação dos objetivos principais.

No capítulo Conceitos Fundamentais, são introduzidos alguns conceitos relevantes para o desenvolvimento do trabalho, abordando os cuidados de saúde e vida assistida, com dispositivos médicos destinados à monitorização do batimento cardíaco, temperatura e detecção de quedas.

O capítulo Estado da Arte apresenta o estado da tecnologia para dispositivos *wearable*, fazendo referência a alguns exemplos existentes. Aqui são apresentadas algumas aplicações possíveis, e alguns equipamentos ou protótipos já existentes.

No capítulo Tecnologias e Ferramentas de Desenvolvimento, são introduzidas algumas tecnologias e ferramentas utilizadas ao longo do trabalho.

No capítulo Especificação e Descrição do sistema, é realizada a especificação e descrição do sistema. Os requisitos e os cenários de aplicação explorados são também aqui apresentados.

No capítulo Arquitetura De Hardware salienta-se as principais funcionalidades da plataforma e o seu funcionamento. Também são descritos os principais componentes necessários para o funcionamento de todas as aplicações.

No capítulo Arquitetura De Software são abordados os tópicos de implementação, drivers, sistema operativo, middleware e aplicações.

No capítulo Arquitetura mecânica, é realizado o estudo e desenho mecânico do sistema. Desde a disposição da eletrónica ao design do protótipo.

No capítulo Conclusões, são apresentadas as conclusões sobre o trabalho realizado bem como eventuais trabalhos futuros.

Em algumas imagens o texto encontra-se escrito em inglês para facilitar a contextualização.

2. CONCEITOS FUNDAMENTAIS

Neste capítulo pretende-se apresentar alguns conceitos fundamentais relativos ao trabalho, estes são importantes para a compreensão das opções tomadas no desenrolar do projeto.

São vários os conceitos abordados ao longo deste trabalho, uma vez que se trata de toda a estrutura da solução, sendo então apresentados alguns dos conceitos mais relevantes relacionados com a visão do projeto e a própria implementação do trabalho desenvolvido.

2.1. DISPOSITIVOS *WEARABLE*

2.1.1. MODOS DE OPERAÇÃO

Os modos de operação definem de que forma o utilizador e as máquinas devem interagir.

- **Interação**

Por norma a interação entre o utilizador-computador é associado ao conceito de sessão, tendo em conta que toda a interação acontece entre o momento em que é iniciada a sessão até ao momento em que é terminada essa mesma sessão.

Por sua vez os dispositivos *wearable* estão constantemente disponíveis, fazendo com que o fluxo de informações entre o dispositivo e o utilizador, ou vice-versa, seja contínuo. A presença contínua do dispositivo no corpo permite uma adaptação a longo prazo, o que leva a que seja criada uma nova forma de sinergia entre o utilizador e o dispositivo.

- **Aperfeiçoamento**

Os computadores tradicionais são normalmente utilizados para a resolução de problemas específicos e para a comunicação em rede. Em ambos os casos, a utilidade dos computadores está essencialmente ligada à sua eficiência e eficácia na realização dessas atividades. Por outras palavras, pode dizer-se que os computadores tradicionais são projetados com o intuito de realizar as tarefas (computação/comunicação) com o maior desempenho possível.

No que trata aos dispositivos *wearable* assume-se que a principal tarefa a realizar não está relacionada com o processamento de dados mas sim com os objetivos/necessidades do utilizador. Desta forma, os dispositivos *wearable* devem funcionar de forma a potenciar as capacidades do utilizador, através do aumento dos seus sentidos, memória, comodidade entre outros.

- **Mediador**

Os dispositivos *wearable* têm como característica a capacidade de encapsularem o utilizador, e dessa forma funcionar como um mediador entre o utilizador e o ambiente que o rodeia. Daqui nasce uma consequência básica, que é a possibilidade do dispositivo agir como filtro de informações.

Enquanto filtro de informações, os dispositivos *wearable* permitem que o utilizador defina o tipo e a quantidade de informações que pretende receber. Isso permite, por exemplo, que o dispositivo filtre de forma personalizada o tipo de alertas de um telemóvel que serão apresentados ao utilizador de acordo com os seus interesses. Além disso, o dispositivo pode, por exemplo, servir como um agente intermediário, ajudando o utilizador em tarefas do dia-a-dia.

2.1.2. ATRIBUTOS DE UM DISPOSITIVO *WEARABLE*

Os atributos definem as principais características de um dispositivo *wearable*, e podem ser tais como:

- **Não restritivo:** A sua utilização não impede o utilizador de realizar outras tarefas
- **Facilidade de interação:** O interface com o utilizador tem de ser simples e intuitivo.
- **Controlável:** Tem de ser possível ao utilizador controlar a forma e a quantidade das informações disponibilizadas.
- **Monitorização do ambiente:** O ambiente em redor do utilizador é constantemente monitorizado, sendo extraídos apenas os dados relevantes definidos pelo utilizador.
- **Capacidade de comunicação:** O dispositivo expande a capacidade de comunicação do utilizador com o ambiente que o rodeia.

2.2. CUIDADOS DE SAÚDE E DE VIDA ASSISTIDA

O corpo humano é constantemente submetido a vários tipos de stress positivo e/ou negativo no dia-a-dia. Atividades desconfortáveis, sons barulhentos, doenças e os mais diversos perigos provocam stress negativo para o corpo.

Nos últimos anos, a importância dada pelas pessoas a uma vida saudável tem vindo a aumentar. Como tal, tem havido um crescente interesse no desenvolvimento de produtos de monitorização da saúde de forma precisa, eficiente e compacta. Nos cuidados de saúde, vários índices são utilizados para determinar a condição de saúde, entre eles: batimento cardíaco, temperatura e humidade corporal e deteção de quedas. Estes são explicados sucintamente de seguida.

2.2.1. BATIMENTO CARDÍACO

Ao registo da atividade elétrica no coração dá-se o nome de Eletrocardiograma (ECG). O princípio básico do eletrocardiograma é a captação, ao nível da superfície cutânea, dos potenciais elétricos gerados pela atividade do músculo cardíaco. Num coração saudável

esta atividade de contração e relaxamento realiza-se de uma forma coordenada e cíclica, criando aquilo a que se chama batimento ou ciclo cardíaco.

Para se entender o ciclo cardíaco e os mecanismos de sincronismo associados, é preciso, antes de mais, descrever os dois fenómenos que acontecem durante o batimento cardíaco. O primeiro é um potencial elétrico que é gerado pelo nodo sino-arterial e o segundo é o potencial mecânico gerado pela contração ordenada do miocárdio. No primeiro evento ocorre um disparo de atividade elétrica que rapidamente se dissipa por cada célula mio cárdica, o segundo evento é a resposta do conjunto de células mio cárdicas que produz a contração e por conseguinte o movimento de sangue dentro de cada câmara cardíaca.

2.2.2. TEMPERATURA / HUMIDADE CORPORAL

Os seres humanos são homeotérmicos ou seres de sangue quente, tendo a aptidão de regular a temperatura do corpo em vez de ela ser ajustada pelo ambiente externo. A conservação de uma temperatura constante é muito importante para a hemóstase, que é um conjunto de mecanismos que o corpo humano usa para parar uma hemorragia. A maior parte das enzimas são termo sensíveis e só desempenham as suas funções dentro de intervalos de temperatura reduzidos. A temperatura ambiente é demasiado baixa e variável para o normal funcionamento enzimático, logo o calor produzido pelo metabolismo ajuda a manter a temperatura corporal estável e especialmente elevada para as reações enzimáticas.

Considera-se normal uma temperatura média de 37°C. A temperatura rectal aproximasse da verdadeira temperatura central, contudo existem algumas variações ao longo do dia. Essa variação, que não deve ser superior a 0,6°C, é mínima de manhã e máxima à noite. A temperatura na superfície da pele resulta de um balanço térmico que conta com a temperatura interna da pessoa e a temperatura do meio envolvente.

2.2.3. DETECÇÃO DE QUEDAS

Uma queda pode ser definida de várias maneiras, depende do objeto de estudo. O caso considerado nesta análise é a queda de uma pessoa. Uma das definições de uma queda é um “evento não intencional que tem como resultado a imobilidade seja ela no chão ou noutro nível”. [16]. Em termos anatómicos, uma queda ocorre ao longo de dois planos: sagital (eixos X-Z), que separa o corpo em esquerda e direita, Figura 3(a), e coronal (eixos Y-Z) que divide o corpo em anterior e posterior, Figura 3(b).

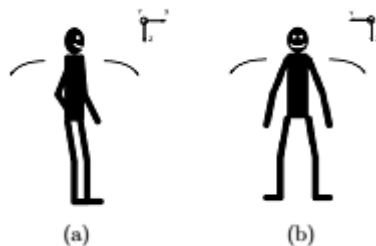


Figura 3 Tipo de planos: Sagital (a), Coronal (b)

A perda de equilíbrio corporal é uma das causas para uma queda, podendo ser consciente ou inconsciente. Durante a queda de uma pessoa, partindo de uma posição estacionária a uma altura, toda a energia potencial é transformada em energia cinética. Durante o impacto ambas as energias anulam-se, pois toda a energia é absorvida pelo corpo. No entanto, se a pessoa estiver consciente a energia resultante pode ser absorvida por alguns músculos usados para amparar a queda, braços ou pernas, Figura 4 (a). Caso se trate de um impacto inconsciente pode resultar em ferimentos graves, Figura 4(b)

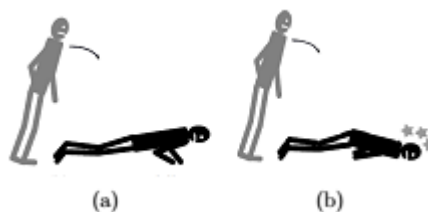


Figura 4 Tipo de quedas: Consciente (a), Inconsciente (b)

Relacionado com a queda está a postura. A configuração do corpo é mais uma variável a considerar, num possível algoritmo de detecção de quedas. A postura é importante pois adiciona informação acerca da posição do corpo, se está deitado, sentado, encostado ou até mesmo dobrado.

Os fatores de risco associados à probabilidade de uma queda são:

- Intrínsecos: Idade; Fragilidade óssea; Fraca mobilidade; Problemas de visão; Doença de Parkinson; Outras doenças cognitivas;
- Extrínsecos: Individuais (uso de sapatos inadequados); Uso de drogas;
- Ambiente Interno: Escadas; Chão escorregadio;

- Ambiente Externo: Calçadas danificadas; Pouca luminosidade; Locais lotados.

A maior parte das quedas ocorre durante as atividades do dia-a-dia que envolvam uma perda de equilíbrio. Outro tipo de movimentos de rotina como subir escadas ou tentar alcançar objetos em locais elevados elevam a probabilidade de cair. Contudo, em atividades como correr ou exercício físico a probabilidade diminui [4].

3. ESTADO DA ARTE

Existem muitas formas dos dispositivos *wearable* melhorarem tanto a vida cotidiana como o trabalho das pessoas, neste capítulo, inicialmente vão ser introduzidos alguns conceitos básicos diretamente ligados ao conceito em desenvolvimento, depois vão ser apresentadas algumas das aplicações utilizadas comercialmente e finalmente alguns exemplos de dispositivos específicos de algumas empresas.

3.1. CONCEITOS

3.1.1. ASSISTENTE INTERATIVO

O termo de assistente interativo na área da informática está normalmente associado a um apoio no ato da pesquisa. Estes normalmente estão relacionados com a internet, tendo em conta o auxílio que podem dar ao utilizador na realização de melhores e mais completas pesquisas, por exemplo o Google.

Pegando neste mesmo conceito mas associado aos dispositivos *wearable*, pode ser obtido um impacto significativo sobre a forma como trabalhamos em geral: agentes inteligentes contextualmente conscientes que nos podem alimentar com informações em tempo real ou mensagens pessoais, tendo em conta a envolvente em que o utilizador se encontra.

Com a evolução deste tipo de dispositivos, a monitorização das ações do utilizador será constante, fazendo com que o dispositivo passe a agir de forma adequada (por exemplo, um *wearable* não deve interromper as ações do utilizador). Além disso, este dispositivo deve ser capaz de apresentar informações de uma forma que seja apropriada para o utilizador: antes de conduzir um veículo automóvel pode ser interessante visualizar o mapa e ter conhecimento da distância a ser percorrida, no entanto, durante a viagem, é preferível que o dispositivo dite as direções (por exemplo "vire à esquerda no próximo cruzamento") ou, discretamente, apresente setas com as direções no próprio mapa, similar ao atual GPS. Em última análise, um dispositivo *wearable* irá fornecer as informações mais relevantes da forma mais correta, no lugar apropriado e na hora certa.

3.1.2. REALIDADE AUMENTADA

A realidade aumentada associada a dispositivos *wearable* é um tópico muito relevante, especialmente quando se fala em dispositivos que permitem ver o mundo com informações importantes associadas em tempo real. Tomando o exemplo de um operário que necessita de aceder constantemente ao manual ou instruções de um dado equipamento: seria muito mais simplificado se ele tivesse acesso em tempo real a essa informação através de um ecrã frontal ao olho. De modo a que esta informação fosse a mais apropriada o operário utilizaria uma camara montada num capacete e o sistema interpretaria os sinais. Isto permitiria a um operário desempenhar com muito mais eficiência as tarefas de manutenção. Este conceito assenta sobre um dos principais objetivos dos dispositivos *wearable*: a possibilidade de melhorar as capacidades do utilizador sem restringir os seus movimentos.

3.1.3. RECONHECIMENTO DO CONTEXTO

Os dispositivos *wearable* permitem recolher diversas de informações do ambiente e do utilizador através de sensores. Os tipos de sensores podem variar, desde simples sensores de temperatura até conjuntos de sensores com dados complexos que necessitam de ser agregados e interpretados.

O objetivo principal é encontrar métodos de reconhecimento de atividades efetuadas pelo utilizador. A interação entre o utilizador e o dispositivo em desenvolvimento pode ser implementada após o sistema ser capaz de reconhecer este tipo de informação. Os métodos atuais de reconhecimento utilizam técnicas de Inteligências Artificial, no entanto, as mais

recentes investigações utilizam a Aprendizagem Supervisionada. Este conceito baseia-se na análise de dados em bruto para extração de características e preparação de conjuntos de teste, sendo que o objetivo é utilizar a aprendizagem não supervisionada para reconhecimento do contexto. Este tipo de aprendizagem pode permitir o desenvolvimento de novos métodos, estes podem ser associados a vários domínios de aplicações porque possibilitam a aprendizagem de diferentes contextos autonomamente. Ashbrook e Starner apresentam um exemplo para estes métodos de aprendizagem utilizando dados GPS.

3.2. APLICAÇÕES

3.2.1. FORMAÇÃO E SUPORTE

Os dispositivos *wearable* podem disponibilizar informações sobre tarefas específicas ou mesmo assistência quando o utilizador mais necessita. O fato de os enfermeiros estarem constantemente a alterar as suas rotinas, implica uma formação/suporte constante, o que pode tornar-se ineficaz, comparado com a possibilidade dessa formação ser disponibilizada constantemente de uma forma simples e prática. Para tal basta alterar a abordagem da formação/treino para um suporte constante aos enfermeiros enquanto desempenham as suas tarefas.

Integrar as muitas tecnologias educacionais ou de suporte num único sistema que se encontra diretamente associado às ferramentas de trabalho poderá ser um grande desafio, no entanto será uma forma de demonstrar a potencialidade deste tipo de sistemas.

3.2.2. APLICAÇÕES DE ASSISTÊNCIA PARA DESABILITADOS

Algumas aplicações *wearable* para pessoas que sofrem de deficiência física, cognitiva ou sensorial têm sido utilizadas, desde aplicações portáteis, até próteses. Exemplos típicos são as aplicações de guias para invisuais, tais como o VibraVest, que disponibiliza um aviso sobre os objetos mais próximos. Este aviso utilizando o tato, também pode ser usado para ajudar surdos. A complementar o aviso com o tato, os avisos por som e voz também têm sido utilizados para ajuda na orientação de cegos.

Dispositivos que disponibilizam informações, juntamente com a tradicional vara, um cão-guia e sons ambientais, têm sido utilizados para ajudar invisuais, permitindo a deteção de

proximidade de pessoas, animais e objetos. A identificação por RFID é também utilizada para a orientação de cegos e indicação do caminho que deverão seguir.

As aplicações acima foram desenhadas para ajudar os aleijados diretamente. No entanto existem ocasiões em que o assistente é necessário para permitir que os outros comuniquem com os aleijados.

3.3. EQUIPAMENTOS / PROTÓTIPOS

Os dispositivos *wearable* foram abraçados já algum tempo por algumas organizações, utilizando-os de formas inovadoras de modo a melhorar a eficiência e a qualidade nos seus processos. Mais recentemente algumas empresas inovaram ao disponibilizar os estes sistemas para tarefas comuns do quotidiano.

3.3.1. MYO (GESTURE ARMBAND CONTROLLER)

O MYO trata-se de uma banda “elástica” que é colocada no antebraço e que é capaz de ler os impulsos elétricos que movimentam os músculos. O sistema consegue detetar a intenção do movimento (quando a ordem inicial chega aos músculos) mesmo antes do músculo comece a reagir. Para além disso, conta ainda com acelerómetros e giroscópios para detetar movimentos das mãos e dos dedos. O objetivo é permitir a interação com computadores ou outros dispositivos digitais de recurso a fios através do Bluetooth. Na Figura 5 encontra-se o aspeto do dispositivo [5].



Figura 5 MYO

3.3.2. BATIMENTO CARDÍACO ATRAVÉS DO SOM

O som que ouvimos da caixa torácica inclui o som dos pulmões e o som do coração. Os sons provocados pelos movimentos do coração durante um ciclo cardíaco são geralmente descritos como *lub-dub* ou *lub-dup*. O primeiro som é causado pelo fecho das válvulas atrioventriculares e o outro é provocado pelo fecho das válvulas semilunares.

Apesar da facilidade atual da monitorização do batimento cardíaco (através de estetoscópio e eletrocardiograma), este procedimento ainda está muito restrito a consultas medicas ou visitas às farmácias. O estetoscópio e o eletrocardiograma passaram a ser o principal meio através do qual são detetadas e diagnosticadas doenças cardíacas.

A ausculta dos sons do coração pelo estetoscópio convencional é simples e não invasiva, no entanto, um dos principais problemas deste método é que é uma medida bastante subjetiva. Mais recentemente, o batimento cardíaco, também passou a poder ser monitorizado através da medição das taxas de impulsos, em qualquer ponto do corpo onde a pulsação da artéria fosse transmitida para a superfície. No entanto continua a não ser conveniente para as pessoas a utilização destes equipamentos no dia-a-dia. Na Figura 6 encontra-se o processo para obter o batimento cardíaco através do som.

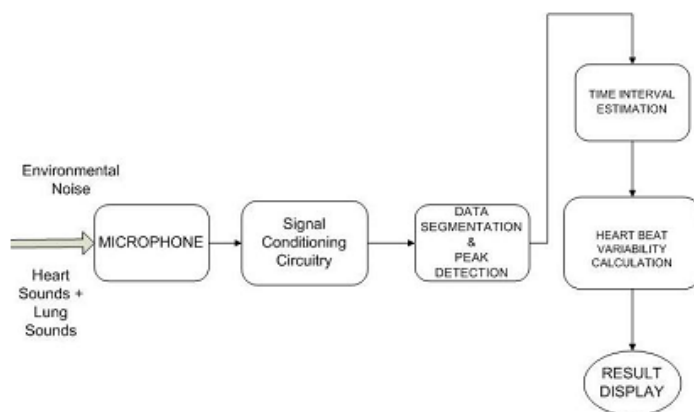


Figura 6 Batimento Cardíaco Através do Som

Outra forma de obter o batimento cardíaco é através do princípio de funcionamento de um estetoscópio, ou seja, medir o batimento cardíaco através do som. Este é constituído por um microfone capaz de detetar sinais bioacústicos associado a um pequeno circuito

eletrónico e um PDA. Permitindo assim monitorizar o batimento cardíaco de uma forma não invasiva, no entanto todos os sons envolventes podem prejudicar as medidas [3].

3.3.3. GIST(WEARABLE GESTURAL INTERFACE FOR REMOTE SPATIAL PERCEPTION)

A percepção espacial é uma tarefa desafiadora para as pessoas cegas devido à limitação em interagir com o espaço envolvente. Para tentar resolver este problema, existe o GIST, um equipamento que é uma *interface wearable*, e permite oferecer a capacidade de percepção espacial através da transformação das mãos do utilizador em sensores. Para isso é usado uma camara *wearable* que executa a função de sensor de profundidade, informando o utilizador sobre o espaço em seu redor, dependendo dos diferentes gestos que este faça. Ao permitir os utilizadores cegos explorar o espaço físico através de gestos, o equipamento aproxima a realidade aumentada da realidade física, o que facilita a interação espacial [9].

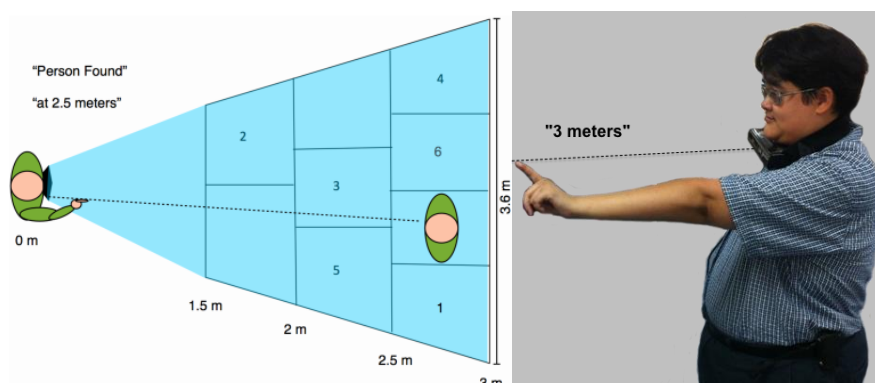


Figura 7 GIST

Através de uma camara *wearable*, a Kinect, o GIST analisa o espaço físico visível e permite aos utilizadores cegos acederem à informação espacial, usando gestos diferentes. O GIST suporta quatro tipos de funções para percepção espacial:

Sensor de cor: O gesto em “V” usando o apontador e o dedo do meio, funciona como uma ferramenta de cor. O apontador e o dedo do meio formam um quadrado, e a cor dominante no meio é amostrada. Os utilizadores podem diminuir ou aumentar a distância entre os dois dedos, de forma a permitir que a área de análise de cor seja maior ou menor.

Presença Humana: o gesto do pulso, permite procurar a presença humana numa determinada área.

Profundidade apontada: apontando com o dedo indicador devolve a distância a que está o objeto a ser apontado. Uma área de 20 por 20 pixéis á volta do centro do dedo é amostrada para informar sobre a média da profundidade. Esta distância é informada em metros ou pés, conforme as preferências do utilizador.

Profundidade na área: o gesto de mão aberta com os cinco dedos é uma extensão para ativar o sensor de profundidade numa determinada área. Em vez da área definida de 20 por 20 pixéis, como no modo anterior, neste é o utilizador que define a área a ser amostrada. O GIST irá informar a profundidade mais pequena nesta área. Este gesto pode ser útil para encontrar objetos que não conseguem ser encontrados através da cor, como por exemplo uma chávena branca numa mesa branca.

3.3.4. SONY SMARTWATCH 2

O Sony SmartWatch 2, também conhecido como SW2, é um dispositivo *wearable* lançado no final de setembro deste ano (2013). O relógio tem um corpo de alumínio onde se encontra o ecrã LCD transflectivo com uma resolução 220x176 pixéis. O SW2 liga-se ao telemóvel utilizando o Bluetooth, tendo como suporte o NFC para facilitar o emparelhamento [6].

- Especificações principais:
- Ecrã: LCD transflectivo de 4cm com 220 x 176 pixéis
- Processador: single-core de 280MHz
- Sistema Operativo: Variante modificada do Android
- Conectividade: Bluetooth 3.0, NFC, micro USB
- Dimensões: 42 x 9 x 41 milímetros
- Peso: 122.5g
- Resistente à água: IP57



Figura 8 Sony SmartWatch 2

3.3.5. JAWBONE UP

O Jawbone UP é um dispositivo com a função de pedómetro e monitor de atividade, que emparelhado com uma aplicação no telemóvel, permite criar um registo da atividade física, qualidade do sono e dieta. Além disso, é possível ao utilizador definir alertas para quando a sua atividade física está abaixo de um determinado nível, avisando-o que está parado á demasiado tempo através de alertas vibratórios.

Para aceder á informação guardada é necessário ligar o dispositivo a um equipamento compatível através da entrada de áudio, uma vez que este não tem uma interface que permita interpretar a informação guardada, e iniciar a aplicação UP para sincronizar e ver os dados gravados [7].



Figura 9 Aspeto do Jaybone UP

3.3.6. MISFIT SHINE

O Misfit Shine é um monitor de atividade física, com um corpo em metal que sincroniza com o telemóvel sem qualquer tipo de fios, apenas basta coloca-lo em cima do visor. Os algoritmos deste dispositivo conseguem detetar atividades como ciclismo e natação, para além das mais comuns, como caminhar e correr.



Figura 10 Aspeto do Misfit Shine

Para visualizar o quanto ativo se está durante o dia, basta carregar em cima do dispositivo. Quanto mais ativo se estiver a ser, mais luzes aparecem através de pequenos buracos presentes [8].



Figura 11 Misfit Shine a indicar a atividade diária

A bateria tem uma autonomia de seis meses, e pode ser trocada pelo utilizador. As suas principais características são:

Tamanho inferior ao de duas moedas juntas de 50 cêntimos.

- À prova de água
- Técnica de proprietário de transmissão de informação, para qualquer dispositivo iOS5, que não usa cabos, WiFi ou Bluetooth.
- Para sincronizar, apenas é necessário colocar o dispositivo em cima do ecrã do telemóvel.



Figura 12 Posição do Misfit Shine para sincronizar

4. TECNOLOGIAS E FERRAMENTAS DE DESENVOLVIMENTO

Neste capítulo descrevem-se as tecnologias e as ferramentas de desenvolvimento adotadas ao longo do trabalho. São apresentadas as linguagens de programação utilizadas, assim como os conhecimentos considerados necessários.

4.1. EM::BLOCKS (C/C++ IDE)

O *Em::Blocks* [10] é um software livre e de código aberto, criado para responder às diversas necessidades dos utilizadores que trabalham na área de desenvolvimento de software embestado. Este software é bastante extensível e totalmente configurável, apresentando um estilo semelhante ao *Visual Studio*, sendo no entanto baseado em *CodeBlocks* [11] e no *GCC* [12].

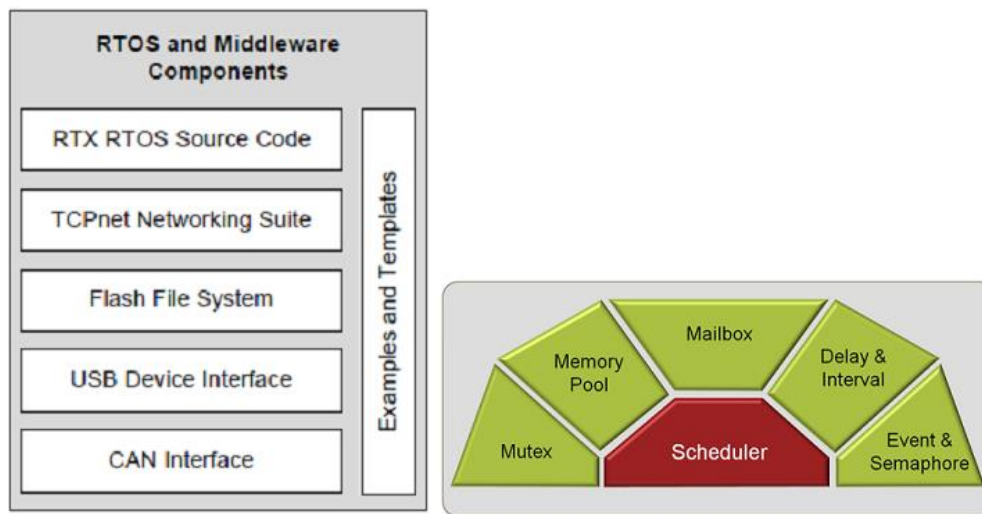


Figura 14 Arquitetura do RTOS RTX

As principais características do RTX incluem:

- É um RTOS *royalty-free*
- Programação flexível: round-robin, preventiva e de colaboração
- Operações em tempo real com baixa latência de interrupção
- Tamanho reduzido para sistemas com recursos limitados
- Número ilimitado de tarefas, cada uma com 254 níveis de prioridade
- Número ilimitado de filas de mensagens, semáforos, mutex e temporizadores
- Suporte para multithreading e operações thread-safe

4.3. ALTIUM DESIGNER

O Altium Designer é um *software* de desenvolvimento utilizado para a criação de placas de circuito impresso. Este permite desenhar o esquemático e posteriormente importar o esquema para a placa de circuito impresso. Uma das suas principais vantagens é a capacidade de incorporar o modelo 3D dos componentes, o que é importante quando se está a lidar com espaços extremamente reduzidos, algo inerente a um dispositivo que se considere *wearable*.

4.3.1. ESQUEMÁTICO

O esquemático é a parte do ambiente de desenvolvimento onde se criam as ligações elétricas. Para tal é necessário colocar os símbolos do esquemático e efetuar as ligações pretendidas. Na imagem que se segue é mostrado uma pequena parte do esquemático

criado para o protótipo do equipamento. Aqui pode-se visualizar alguns componentes que foram criados, e as respectivas ligações.

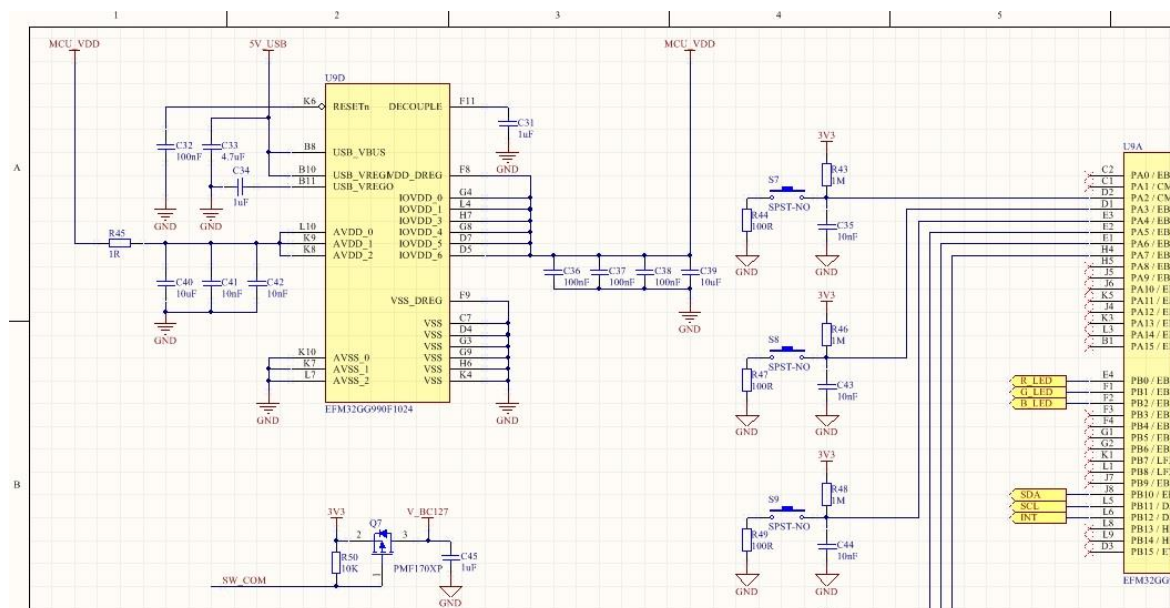


Figura 15 Esquema elétrico

4.3.2. DEFINIÇÃO DE MODELO 3D

O modelo 3D dos componentes tem que ser criado aquando da criação do componente. Este deve ser o mais fiel possível, seguindo fielmente os valores fornecidos pelo fabricante. No exemplo que se segue está o modelo do microcontrolador.

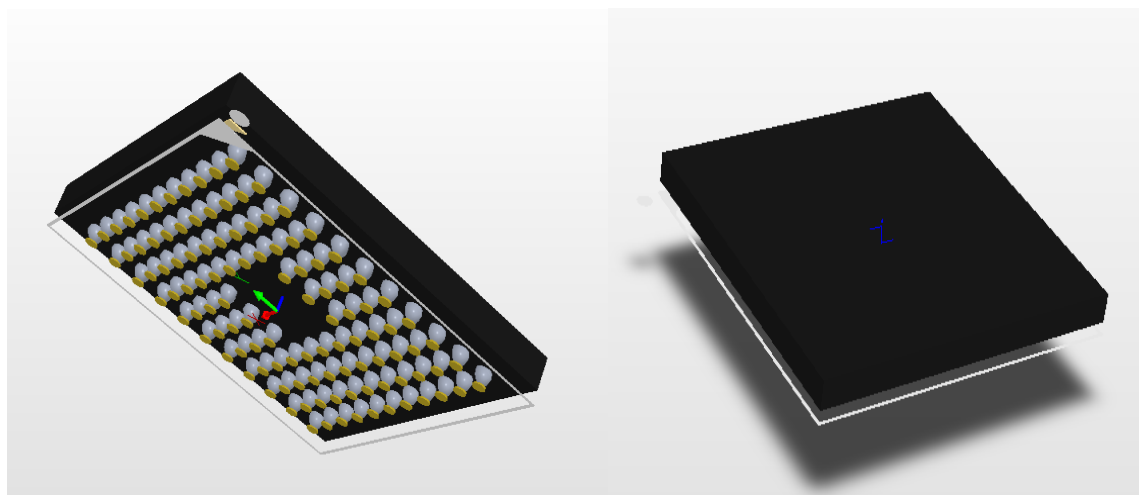


Figura 16 Modelo 3D do microcontrolador

4.3.3. PCB

Após a criação de todos os componentes e da prévia definição do esquemático, é necessário colocar os componentes que tem interface humana, ou cuja posição já foi definida por restrições mecânicas. Após esta colocação são realizadas as ligações de pistas, entre todos os componentes. No fim de várias iterações obtêm-se um desenho de uma placa de circuito impresso, que posteriormente pode ser fabricada.

4.4. SOLIDWORKS

A mecânica está presente não apenas como uma vertente estrutural do equipamento mas também de design e de organização dos equipamentos eletrónicos. Deste modo, a componente da mecânica revelou-se um aspeto importante para apresentar neste trabalho, pois é através desta que é possível criar o interface entre o equipamento e os utilizadores.

Note-se que o método de fabrico não é uma preocupação nesta fase, sendo o modelo desenvolvido apenas para demonstração do aspeto final aproximado. Uma das preocupações iniciais, e que deve estar presente ao longo do desenvolvimento mecânico, é a sua adaptabilidade aos diversos pulsos, ainda é interessante enveredar por uma solução que permita variar o aspeto do equipamento. Assim, sugere-se que o equipamento permita remover a pulseira para permitir uma variação de cor ou mesmo de material desta.

Nesta fase, o principal interesse encontra-se em transmitir o aspeto visual desejado para o equipamento, para tal foram realizados alguns desenhos preliminares à mão, passando para um *software* de desenho mecânico - *SolidWorks*.

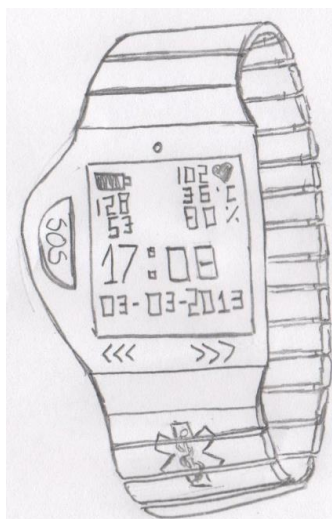


Figura 17 Desenho preliminar

Tornou-se um desafio aprender um programa de uma área completamente nova, no entanto revelou-se uma ferramenta crucial para traduzir os componentes eletrónicos num equipamento comercial. O primeiro passo para a obtenção do objeto a três dimensões é começar por um simples desenho bidimensional representativo da Figura 17, e adicionar-lhe comprimento (ferramenta Sketch e Extrude) [13] .

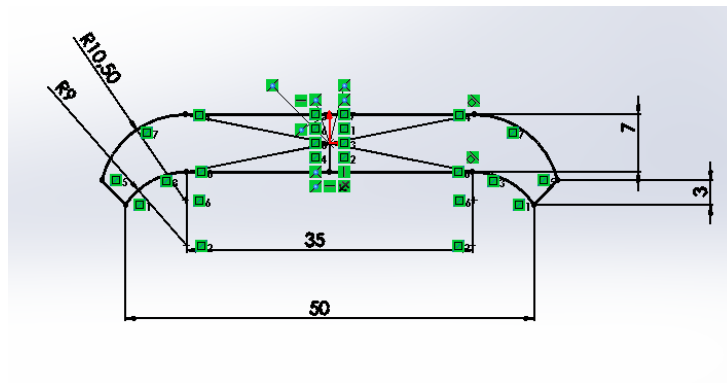


Figura 18 Ferramenta Sketch

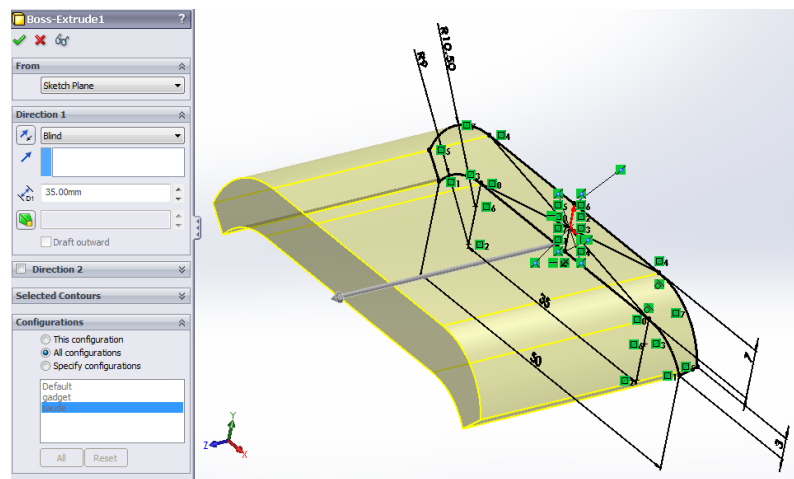


Figura 19 Ferramenta Extrude

Outra ferramenta do SolidWorks é o CutExtrude, este permite remover material, criando indentações e espaços vazios para a colocação da eletrônica. Nesta fase não foi determinada com grande precisão o volume necessário, sendo apenas para demonstração.

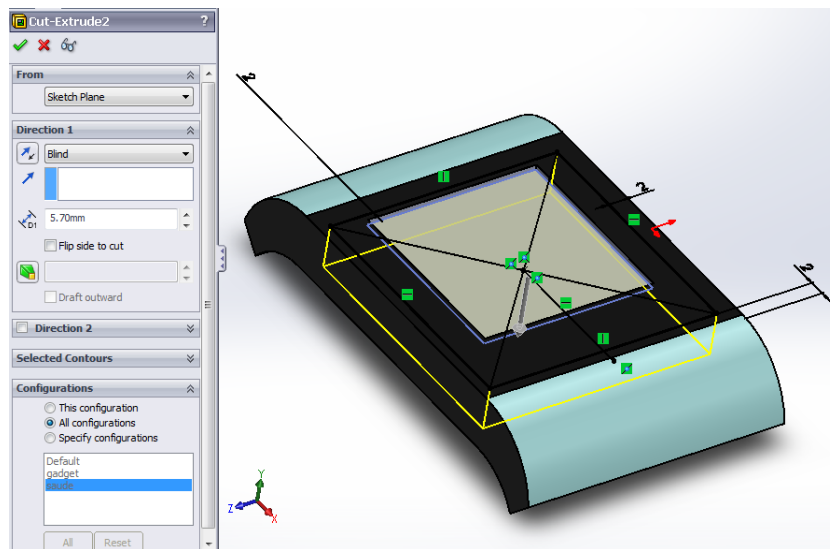


Figura 20 Ferramenta CutExtrude

Para avaliar a disposição do exterior modelou-se a três dimensões alguns componentes exteriores como o ecrã, proteção do ecrã, slide e botões. Estes foram obtidos através do Extrude segundo as dimensões dos componentes seleccionados.

Por fim, recorreu-se a uma outra ferramenta *Appearances* através da qual se procede à seleção do material, textura e cor do equipamento. Para a obtenção do modelo final é necessário ‘montar’ o equipamento, para tal faz-se um *Assembly* onde todas são agrupadas de modo a obter o objeto final [13].



Figura 21 Aparência do equipamento desenhado

5. ESPECIFICAÇÃO E DESCRIÇÃO DO SISTEMA

5.1. VISÃO GERAL DO SISTEMA

O sistema proposto pretende responder a um conjunto de situações com um único dispositivo de fácil e cómoda utilização quando colocado no pulso do utilizador.

As aplicações deste tipo de dispositivo vão desde a monitorização de parâmetros relativos ao utilizador tais como:

- Batimento cardíaco
- Oximetria
- Temperatura corporal
- Humidade corporal
- Atividade física
- Monitorização do sono

- Chapa de identificação com dados vitais
- Detecção de quedas

Com o dispositivo poderá também ser possível interagir com outros sistemas nomeadamente com o recurso das facilidades existentes como as comunicações: Bluetooth, WIFI ou possivelmente RF.

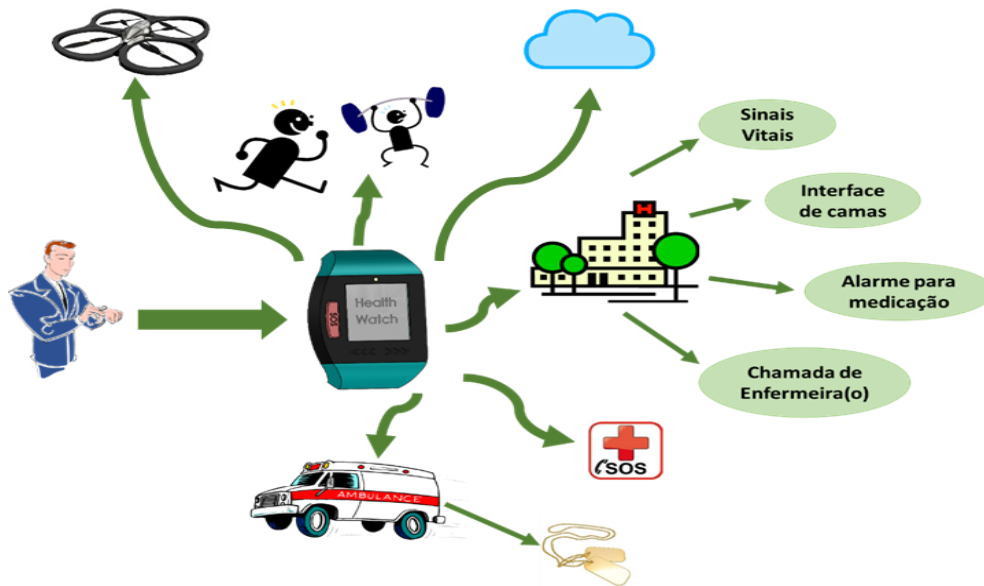


Figura 22 Interação do dispositivo com o mundo

5.2. REQUISITOS GENÉRICOS DO SISTEMA

Este capítulo apresenta um conjunto genérico de necessidades independentes dos casos de aplicação aqui explorados. Os requisitos mais específicos de cada caso de utilização são apresentados junto da explicação desses mesmos casos.

Interface do sistema

A arquitetura do sistema deverá permitir a implementação de interfaces *user-friendly* cujo *design* considere a capacidade de aprendizagem de utilização rápida e interface intuitivo, facilidade em lembrar como se usa, eficiência e produtividade da utilização, tolerância a erros do utilizador. Os interfaces multimodais (exemplos: áudio, visual, toque, toque

deslizante e reconhecimento gestual) também devem ser privilegiados nas diferentes aplicações de forma a criar uma interação mais rica com o utilizador.

Multi-aplicação

O desenho do Sistema deve assentar na decomposição e suportar modularidade para permitir a criação de um ambiente multi-aplicação inclusive cenários *multi-vendor*, permitindo desenho e desenvolvimento independentes de aplicações e subsistemas.

Para tal, a arquitetura deve garantir isolamento de interferências não esperadas entre os diferentes subsistemas integrados.

A especificação da interface da arquitetura tem de permitir a integração de novos subsistemas sem a ser necessário perceber o funcionamento interno dos componentes da arquitetura.

Determinismo temporal

A arquitetura do sistema deverá garantir que o tempo de arranque e execução dos componentes de software é conhecido e limitado, de forma a permitir implementar tolerâncias a falhas, reiniciando o componente após a deteção da falha.

Adicionalmente, a arquitetura deverá garantir a sincronização computacional e de comunicações de serviços distribuídos e validação temporal dos dados gerados em tempo real pelos diferentes subsistemas. Para tal, um serviço de tempo global deve ser distribuído a todos os serviços.

Transferência de mensagens

A arquitetura deverá permitir transferência de mensagens despoletadas por eventos, desenhada tendo em conta uma carga de comunicação média, permitindo falha de comunicação durante cenários de elevada procura de comunicação.

A transferência de mensagens em tempo real também deverá ser permitida, suportando o transporte de mensagens de forma determinística temporal.

A infraestrutura de comunicação física deverá permitir ser compartilhada pelos diferentes serviços da arquitetura.

Gestão de recursos

A arquitetura deverá permitir a escalabilidade dinâmica da performance dos recursos de hardware com base no nível atual de energia disponível, nomeadamente controlo de distribuição de energia para componentes não usados ou até reconfiguração e ajuste da frequência de relógio.

A arquitetura do sistema deverá permitir a reconfiguração tal que possibilite um conjunto alargado perfis de performance a poderem ser usados em diferentes aplicações.

Para a implementação de serviços e aplicações críticas e relacionadas com segurança, a arquitetura garantir a coexistência de mecanismos de alocação de recursos estáticos e dinâmicos e recursos de comunicações mesmo que estes recursos sejam partilhados.

Eficiência energética

Os consumos e eficiência energética são atributos de qualidade bastante importantes, especialmente para os dispositivos portáteis, assim sendo, o desenho de baixo consumo e medição dos próprios consumos deverão também fazer parte da arquitetura do sistema.

A gestão dos consumos deve incluir a adaptação e ajuste de componentes a cada aplicação e performance de um uso específico, para que as aplicações corram com os menores recursos e energia possível. Os consumos de standby devem ser de muito baixo consumo, fazendo parte da gestão dos consumos.

A arquitetura do sistema deverá permitir a recolha e armazenamento da informação da energia fornecida e energia remanescente, consumos atuais, consumos de pico e temperatura. Estas informações devem ser apresentadas com um interface adequado a

utilizadores e desenvolvedores. No entanto, a arquitetura também deverá permitir configurações por defeito, para que o utilizador final não necessite de fazer nada relacionado com a eficiência energética para usar o dispositivo e as diferentes aplicações.

Segurança e confidencialidade da informação

A arquitetura deverá fornecer mecanismos e serviços para garantir os conceitos mais básicos de segurança: integridade, disponibilidade e confidencialidade assim como conceitos mais secundários como autenticação e autorização de acesso ao sistema.

Para garantir a integridade, a arquitetura deverá ter mecanismos para prevenir modificações de hardware e software por pessoal não autorizado. Mecanismos para a prevenção de disponibilização de informação a pessoas ou sistemas não autorizados também deverão ser considerados.

Diagnóstico e teste do sistema

O sistema deverá conter serviços de diagnóstico que forneçam informação consistente sobre o estado do sistema.

A arquitetura, deverá considerar um desenho para a testabilidade, possibilitando uma interface de teste estandardizado para o teste dos componentes durante o funcionamento do sistema.

5.3. CENÁRIOS DE APLICAÇÃO

Nesta secção vão ser apresentados os cenários que vão ser explorados no decorrer deste documento.

5.3.1. BATIMENTO CARDÍACO

A preocupação com o estado de saúde através da sua monitorização é cada vez maior nas atuais sociedades. Atualmente as pessoas compram dispositivos eletrónicos médicos para fins específicos, tais como monitorizar o seu batimento cardíaco. Estes são usados em situações muito pontuais, e provavelmente só após a pessoa se sentir doente, com a

agravante que o equipamento pode não estar disponível no momento necessário, porque estes não são objetos que se transportem com regularidade. Mesmo após o uso do equipamento, a informação por si só indica apenas a condição atual do doente, mas não a evolução do seu estado clínico.

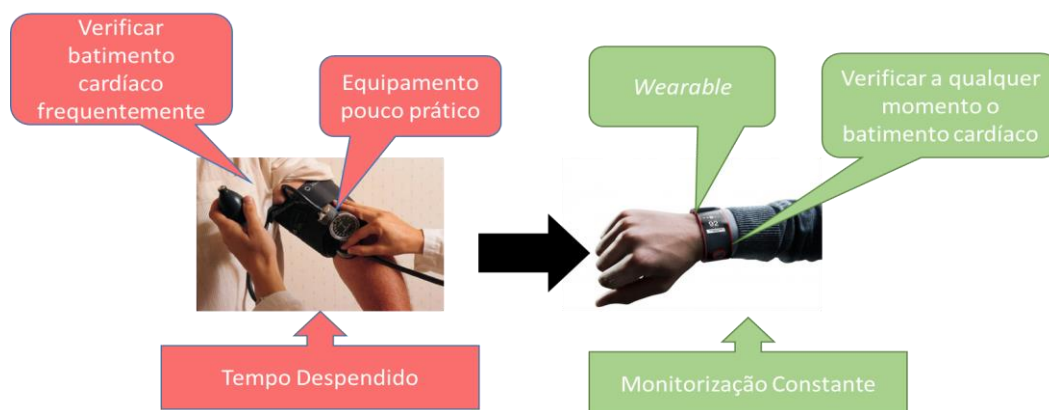


Figura 23 Aplicação do batimento cardíaco

A solução para o problema identificado passa por monitorizar constantemente a pessoa, através de um equipamento que possa ser inserido na sua indumentária habitual, sem causar qualquer transtorno. Para onde quer que a pessoa se desloque, o equipamento está sempre disponível, porque está anexado ao corpo da pessoa. Isto aliado ao facto de ser possível retirar amostras do batimento cardíaco, permite antever uma degradação no estado de saúde do indivíduo, bem como informar um médico da tendência clínica.

Necessidades da arquitetura para monitorização do batimento cardíaco

O sistema deverá permitir a aquisição dos sinais relativos ao batimento cardíaco do utilizador, carecendo também de todas as facilidades de filtragem e agregação desta informação. Esta informação deverá ser passível de ser visualizada no próprio dispositivo relativa a um período de tempo, como ser transmitida de forma segura para uma infraestrutura de armazenamento de dados.

A arquitetura deverá também permitir que possam ser criadas aplicações que usem esta informação de forma transparente como qualquer outra facilidade do dispositivo.

5.3.2. TEMPERATURA E HUMIDADE CORPORAL

A temperatura e humidade corporal são parâmetros fundamentais da saúde humana, estas traduzem o estado atual das defesas do organismo, e a sua atividade biológica. Sempre que alguém se sente doente, a tendência é averiguar se esta ou não em estado febril. Numa situação clínica positiva neste estado, é necessário averiguar a temperatura em intervalos de tempo curtos, o que por si só é um incómodo para o doente e para o cuidador.

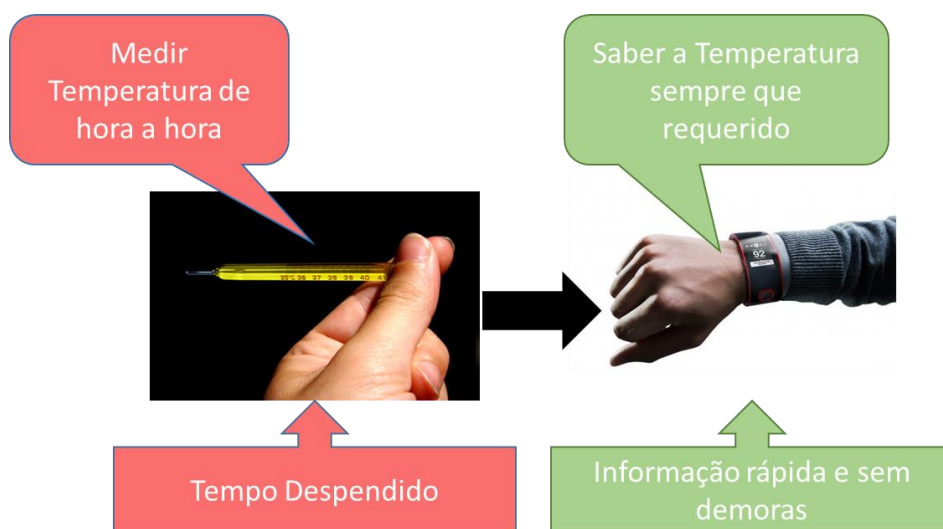


Figura 24 Aplicação da temperatura

Esta tarefa pode ser aperfeiçoada, eliminando a necessidade de fazer estas leituras incomodativas, colocando apenas um dispositivo de fácil adaptação no doente. O objetivo é ter um registo da variação da temperatura, com alertas e visualização do estado atual, permitindo uma maior monitorização e posterior controlo sobre o paciente.

Necessidades da arquitetura para monitorização da temperatura e humidade corporal

A arquitetura do sistema deverá incluir sensores de temperatura e humidade corporal, tendo a capacidade de contextualizar as medições efetuadas. Esta contextualização deverá ser possível com a fusão de outros sensores presentes no sistema permitindo saber se as medidas são válidas ou não.

A configuração de alertas é também uma funcionalidade interessante permitindo avisar o utilizador ou outros sistemas que algo de a temperatura corporal e/ou humidade está fora dos parâmetros normais.

5.3.3. DETEÇÃO DE QUEDAS

As quedas de seres humanos podem ser resultantes de vários fatores e em diversos cenários. Estes cenários podem ser desde atividades desportivas, a contextos domésticos ou de trabalho.

O tempo de resposta do suporte médico numa situação de queda é crítico para mitigação e resolução dos problemas decorrentes da queda, caso contrário as consequências podem resultar em lesões permanentes ou até a morte.

Os fatores envolvidos nas quedas de seres humanos podem ser divididos em:

- Intrínsecos
 - Idade, fragilidade óssea, doenças variadas, problemas de visão, estilo de vida incorrecto (inactividade, obesidade, consumo de álcool).
- Extrínsecos
 - Uso incorrecto de calçado.
- Ambiente Interior
 - Piso escorregadio, escadas.
- Ambiente Exterior
 - Más condições de passeios, fraca iluminação.

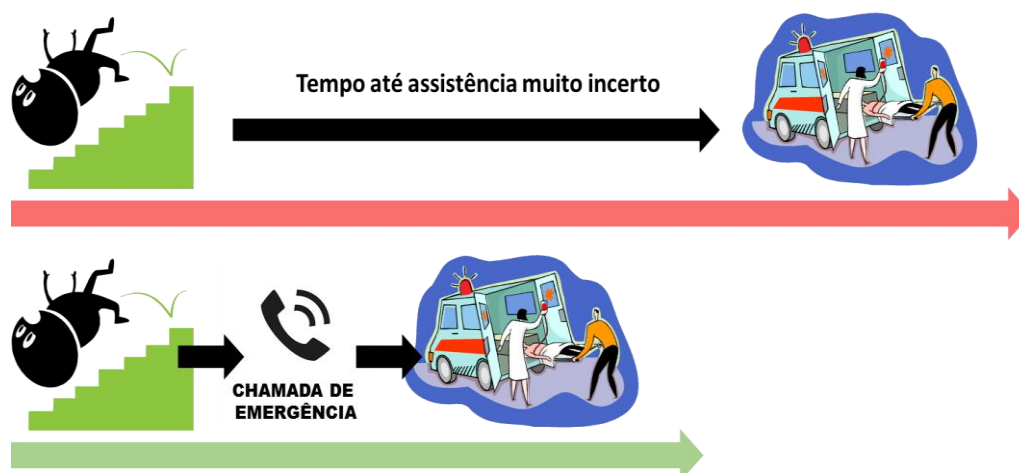


Figura 25 Aplicação de deteção de quedas

Todos estes factores podem contribuir para uma fatilidade, que pode ser minimizada recorrendo a um sistema de monitorização constante. Sempre que um padrão de queda é detectado, pode ser despoletado um alerta de emergência. Neste caso a vítima pode ter auxílio médico de forma mais rápida.

Necessidades da arquitetura para deteção de quedas

A detecção de quedas necessita que o sistema tenha a capacidade de monitorização das acelerações a que o utilizador está sujeito e detecção de acelerações de choque.

O sistema após a deteção de uma queda deverá enviar um alerta em pedido de auxílio, podendo também fazer uma ligação de audio para uma chamada de emergência. Acompanhado do sinal de áudio, um conjunto de informações relativas ao utilizador deveram ser possíveis de transmitir. Caso o dispositivo não tenha disponível as facilidades para gerar o alerta, um sinal audível do tipo buzzer poderá ser gerado.

5.3.1. MONITORIZAÇÃO DE SONO

A qualidade do sono é um parâmetro que pode ser avaliado sobretudo pela atividade dos membros de uma pessoa durante o seu sono. O estudo deste recorre frequentemente a uma monitorização de vários parâmetros fisiológicos durante o sono entre os quais, parâmetros respiratórios, cardíacos, neurológicos, a posição corporal e movimentos dos membros.

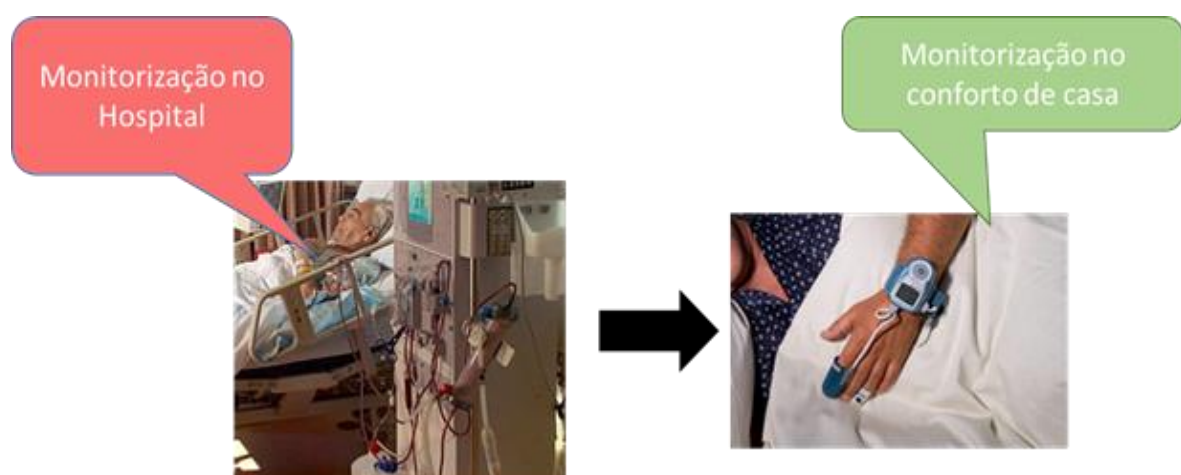


Figura 26 Aplicação de monitorização do Sono

A informação recolhida durante estes estudos, pode ser devidamente tratada, de forma a poder ser incorporada num dispositivo, que indique ao paciente o nível de qualidade de sono da noite anterior. Permitindo ao mesmo que tome medidas durante o dia, de modo a melhorar o seu sono na noite seguinte.

Necessidades da arquitetura para monitorização do sono

A monitorização do sono ou da atividade física em geral recorre à medição das acelerações dos membros do corpo e dos parâmetros cardíacos. Assim a arquitetura deverá conter sensores que permitam medir e inferir acelerações dos membros do corpo e parâmetros cardíacos.

Esta informação deverá poder ser contextualizada com a fusão de outros sensores ou utilizações do dispositivo e depois ser filtradas e agregadas podendo ser visualizadas no próprio dispositivo ou transferida para um outro sistema na nuvem.

5.3.2. CHAPA DE IDENTIFICAÇÃO

As informações vitais de uma pessoa, são todos os dados que são críticos para a sua sobrevivência e bem-estar. Um método para transmissão de informações vitais, é a chapa de identificação de militares, que identifica a pessoa e fornece alguns dados vitais como o tipo de sangue, que podem ser necessários em situações de emergência. Contudo, e dada a evolução da medicina, sabe-se que existem outras informações que podem ser vitais em casos de acidentes, ou emergência médica, onde o paciente não tem capacidade de informar os meios de socorro. Algumas informações vitais são:

- Tipo de sangue.
- Alergias.
- Doenças existentes
- Medicamentos prescritos

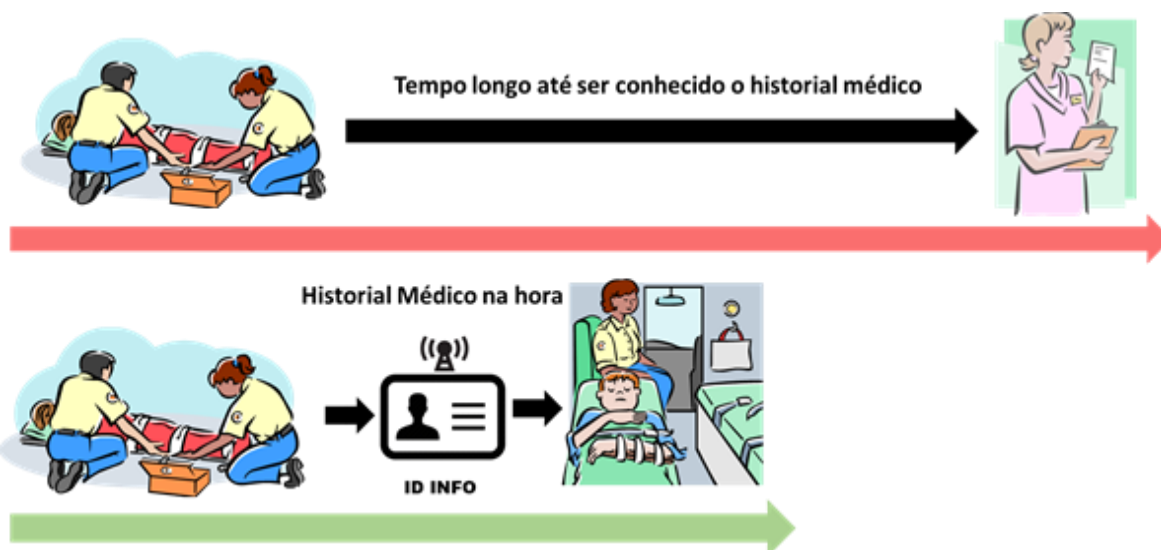


Figura 27 Aplicação da chapa de identificação

Uma vez que esta informação poderia ser algo extensa e incomodativa transporta-la, como o caso de chapas, a solução passa por coloca-la em objetos do quotidiano, que a possam guardar e mostrar a quem de interesse. Assim sendo, aquando de uma emergência médica, os socorristas poderiam ter acesso a essa informação de modo imediato.

Necessidades da arquitetura para chapa de identificação

A incorporação de informações vitais no sistema deverá ser possível. As informações passíveis de serem armazenadas são o tipo de sangue, porte de doenças, alergias e medicamentos continuados.

Questões de autenticidade e autorização de escrita desta informação são um fator fundamental para que esta informação seja utilizada de forma segura. A arquitetura deverá garantir que esta informação é autenticada aquando da adição ao sistema. A confidencialidade dos dados também deverá ser restrita ao uso estritamente necessário e em cenários perfeitamente identificados para o efeito.

6. ARQUITETURA DE HARDWARE

No que refere a um equipamento *wearable*, é de extrema importância a utilização de hardware específico, pois são estes elementos que o caracterizam. Como alguns exemplos pode-se mencionar os elementos de deteção e atuação, armazenamento e processamento de dados, fontes de energia e comunicação são todos igualmente importantes e necessários. As fontes de energia são necessárias para alimentar os componentes que compõe o sistema. A implementação de inteligência e as funcionalidades requerem a capacidade de armazenamento e processamento de dados. Os sensores e atuadores completam o ciclo do sistema de controlo. É através da comunicação que se dá a troca de informação entre todos os componentes do dispositivo, e outros equipamentos, bem como entre o dispositivo e as fontes de informação. A representação base do dispositivo é apresentada na Figura 28.

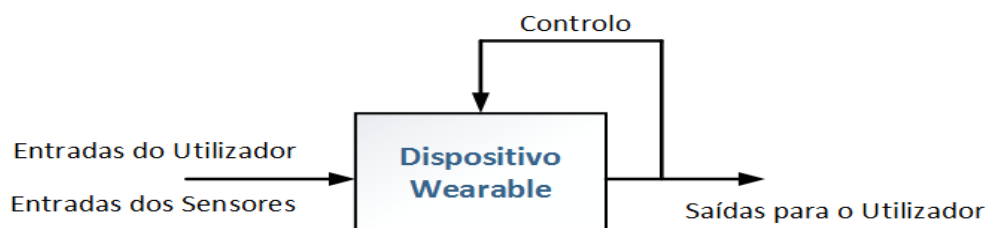


Figura 28 Funcionamento básico do sistema

As entradas de informação para o dispositivo são fornecidas pelo utilizador ou pelos diferentes sensores que monitoriza tanto o utilizador como o ambiente. Tendo este tipo de informação como base da logica de decisão, o sistema proporciona informação ao utilizador e controla o desempenho das funções automatizadas. A seleção dos componentes de hardware para este dispositivo baseiam-se nas aplicações pretendidas e nas condições do ambiente que se pretendem monitorizar. A arquitetura geral do hardware elétrico do dispositivo é ilustrada na Figura 29.

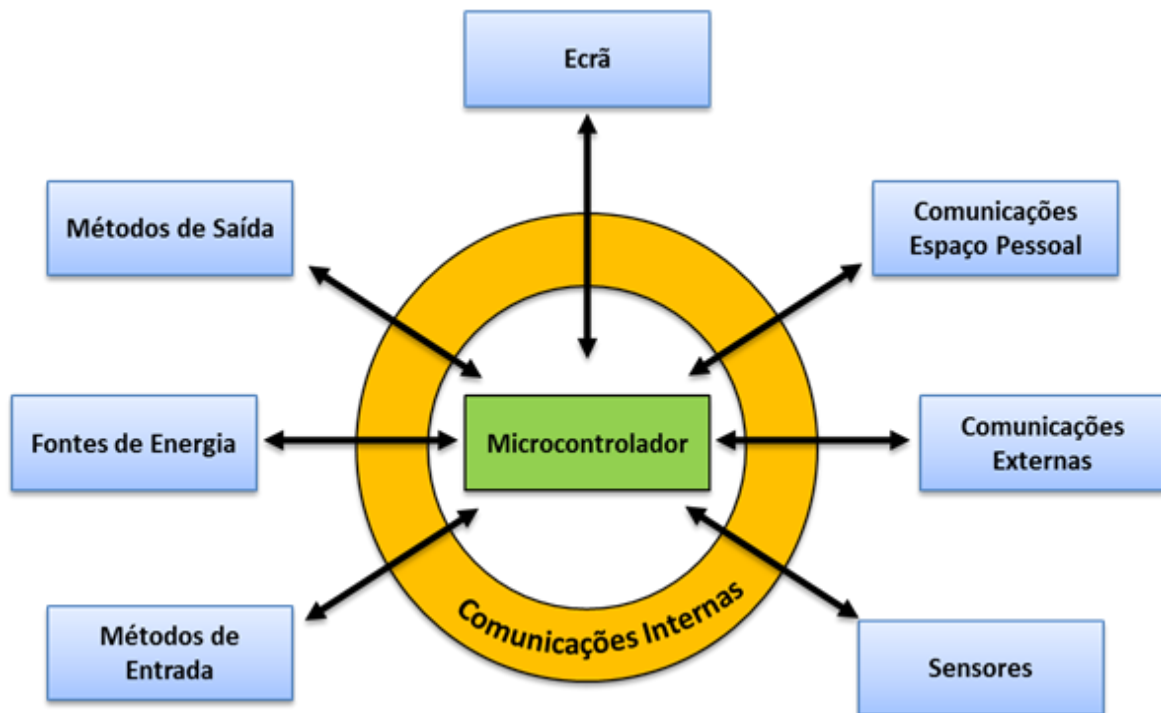


Figura 29 Arquitetura geral do hardware

6.1. MÉTODOS DE ENTRADA

Os métodos de entrada dos dispositivos atuais têm tendência a ter mais interação com o utilizador, uma vez que é necessário serem cada vez mais intuitivos e fáceis de usar. No caso dos dispositivos *wearable*, os métodos de entrada para além de terem que ser intuitivos, devem libertar o utilizador para executar outras tarefas simultaneamente. Exemplos destes métodos são a fala, expressões faciais, movimentos dos olhos, emoções e movimentos corporais.

6.2. MÉTODOS DE SAÍDA

Os ecrãs são o principal método de saída dos equipamentos eletrónicos atuais. No caso dos dispositivos *wearable*, estes desempenham também um papel importante. As pessoas estão familiarizadas com a utilização de ecrãs no dia-a-dia logo este método de saída é natural também em aplicações *wearable*. A aplicação e os dados a serem apresentados determinam o tipo e as características do ecrã a utilizar. Os ecrãs podem ser utilizados de uma forma bastante intuitiva, no entanto, não são adequados para todas as tarefas e são também necessários métodos de saída alternativos e mais diretos.

Além dos métodos de saída para informações complexas como os ecrãs os métodos de saída para notificações simples por exemplo podem passar pela utilização de sinais luminosos (LED's), sinais sonoros (colunas de som) ou mesmo através de sinais vibratórios (vibrador).

6.3. COMUNICAÇÕES

As comunicações nos dispositivos *wearable* são de veras importantes, em primeiro lugar para interligar componentes distribuídos numa área próxima do utilizador (zona pessoal), como por exemplo uma rede de sensores específicos espalhados por todo o corpo. Este tipo de comunicação é considerado interno na medida em que tem lugar dentro da zona de ação do utilizador. Em segundo lugar, a comunicação externa é necessária para a transferência de dados entre o dispositivo *wearable* e outros equipamentos e redes de informação externas. Estas camadas de comunicação são ilustradas na Figura 30.

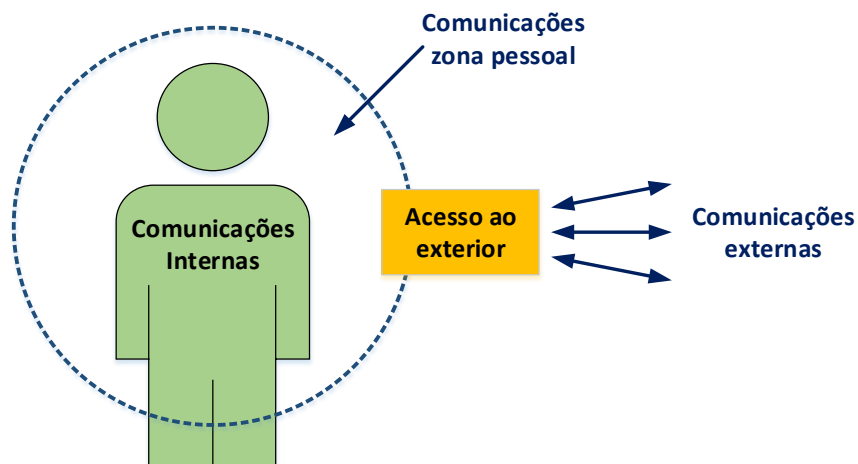


Figura 30 Zonas de comunicação em aplicações *wearable*

Neste modelo de comunicação em existe geralmente um ponto de acesso para permitir comunicações externas. Este ponto de acesso pode, por exemplo, ser uma interface de rede para uma rede de dados moveis (através da ligação a um telemóvel), ou então diretamente à um ponto de rede sem fios.

Para a execução da comunicação estão disponíveis várias técnicas. As técnicas mais adequadas são seleccionados tendo em conta as suas necessidades de comunicação, que são determinadas pelo tipo de dados, as taxas de transferência, periodicidade, fiabilidade, segurança e custo, bem como o consumo de energia.

6.4. FONTES DE ENERGIA

As fontes de energia são deveras importantes em dispositivos *wearable* tendo em conta o importante objetivo da longa duração entre recarregamentos. Além de tecnologias primárias com base em baterias, métodos alternativos de captação de energia são utilizados como por exemplo as células solares. Um dos grandes desafios existentes é razão implícita entre a capacidade energética versus o tamanho necessário. Tendo em conta este desafio, além das evoluções ligadas ao armazenamento da energia, muitas melhorias foram feitas de forma a otimizar o consumo de diversos componentes tanto a nível de hardware como software, entre os quais as unidades de processamento e as unidades de comunicação.

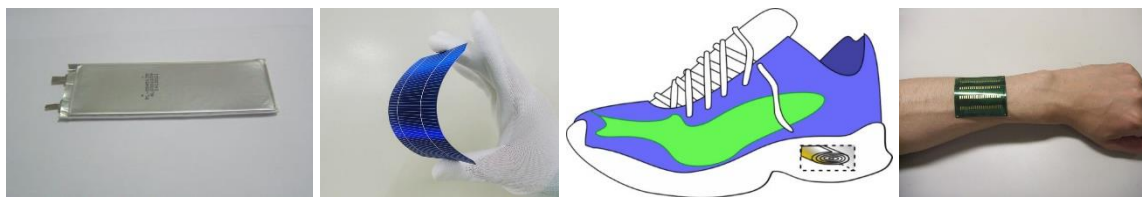


Figura 31 Diferentes fontes de energia

Em ambientes móveis qualquer peso adicional pode prejudicar o conforto de utilização. Normalmente a solução mais comum para aplicações *wearable* é a utilização de baterias de polímero de lítio (Li- polímero), que possibilitam a construção de baterias flexíveis capazes de encaixar naturalmente na estrutura do dispositivo. Outras fontes de energia podem ser obtidas a partir do utilizador ou do ambiente. Por exemplo a utilização de materiais piezoelétricos para a geração de energia através de movimentos do utilizador (como os passos).

6.5. DESCRIÇÃO DOS COMPONENTES

A escolha dos componentes deste trabalho foi um processo demorado, visto que se tenciona conciliar tamanho reduzido, com o menor consumo de energia. Estes parâmetros são cruciais para se obter um equipamento mecanicamente atrativo, e com a melhor autonomia possível.

6.5.1. MÚLTIPLA CONECTIVIDADE

Uma vez que se trata de um dispositivo *wearable*, ter a capacidade de interagir com outros equipamentos é uma mais-valia. Para essa necessidade foi escolhido o Broadcom BCM43341, pois é um integrado que suporta comunicações NFC, IEEE 802.11 a/b/g/n, Bluetooth 4.0, e rádio FM num único elemento. Visto que entrega quatro tipos de comunicações diferentes, este trata-se do elemento ideal para reduzir a área ocupada e o número de componentes necessários. Algumas das suas principais características são:

- Bluetooth 4.0 + EDR.
- Amplificador de potência integrado para Class 1.
- Potência de saída programável, de modo a cumprir os requisitos de potência para Class 1, Class 2, ou Class 3.
- NFC com colheita de energia, para transações mesmo sem energia do dispositivo.
- Tensão de alimentação de até 4.8V.
- Muito baixo consumo.
- WLBGA package.
- Processador ARM® Cortex-M3™ com ROM e RAM integrada.

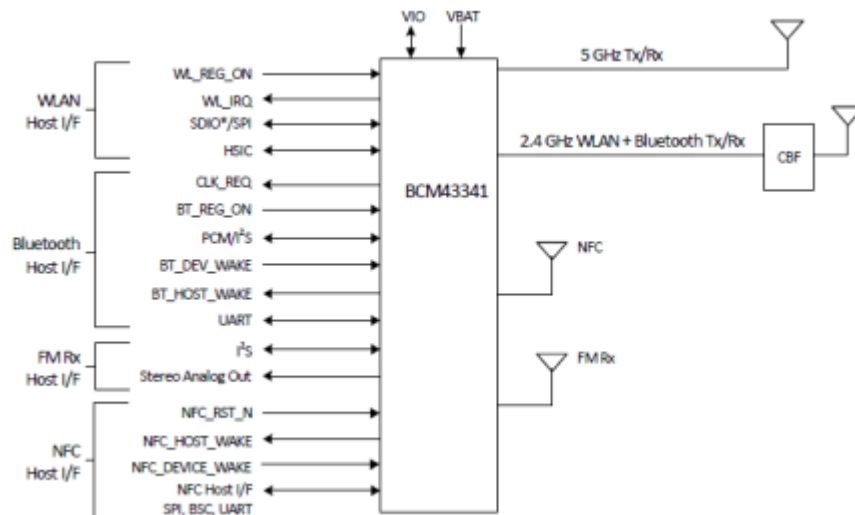


Figura 32 Diagrama de blocos do Broadcom BCM43341

6.5.2. CONECTIVIDADE BLUETOOTH

A comunicação Bluetooth é um *standard* em dispositivos *wearables*, e por vezes, é somente esta a necessária. Como se pretende desenvolver uma plataforma, é imprescindível adequar o hardware às especificações. Para tal, sempre que for necessário ter apenas comunicações Bluetooth e ou ANT, o integrado escolhido é o Panasonic PAN1323, que incorpora três *standards* de comunicações:

- Bluetooth 4.0.
- ANT™.
- Bluetooth Low Energy.

Sendo as características mais atrativas para a plataforma desenvolvida:

- Sem necessidade de componentes externos.
- Dimensão exterior de 9mm x 9.5mm x 1.8mm.
- Custo reduzido face a um integrado de múltipla conectividade.

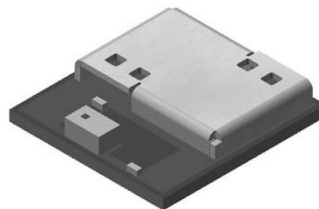


Figura 33 Modelo do módulo de Bluetooth PAN1323

6.5.3. ACELERÓMETRO

Existe uma vasta gama destes equipamentos, desde analógicos, a digitais e integrados. Em uma, duas ou três direções. Para o efeito pretendido este deve ser tri-axial, deste modo foram comparadas as características de vários acelerómetros de modo a escolher o que melhor se enquadrava com o propósito final. As principais características do escolhido são:

- Ampla gama de alimentação, desde 1,7 a 3,7V
- Sensor de temperatura embebido
- LGA-16 package (3 x 3 [mm])
- Máquinas de estado programáveis. Permitem fazer a deteção de algumas situações como quedas, acordar de um estado parado, contar passos, entre outros.

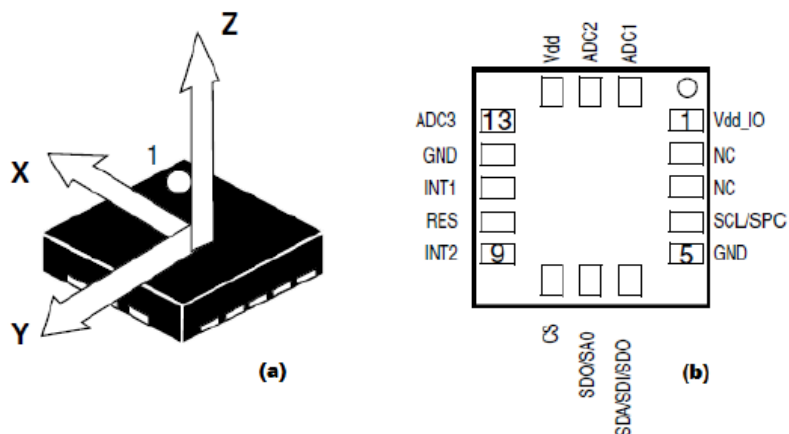


Figura 34 a) Aspeto da caixa do integrado b) Esquema do *footprint* do integrado

6.5.4. MICROCONTROLADOR

O microcontrolador é um componente crítico do sistema, uma vez que é necessário ser rápido o suficiente para tornar a usabilidade o mais fluida possível, e ao mesmo tempo ter um baixo consumo de energia. O microcontrolador que mais se adequa ao trabalho em causa é o Energy Micro EFM32GG990 que conta as seguintes principais características:

- 32bit ARM Cortex-M3 @ até 48MHz
- Consumo de energia de:
 - 20nA @ 3V *Shutoff Mode*
 - 0.4uA @ 3V *Shutoff Mode* com RTC

- 200 $\mu\text{A}/\text{MHz}$ @ 3 V Run Mode
- 1024KB Flash
- 128KB RAM
- USB 2.0
- Package BGA 112, com a dimensão de 10x10 [mm]

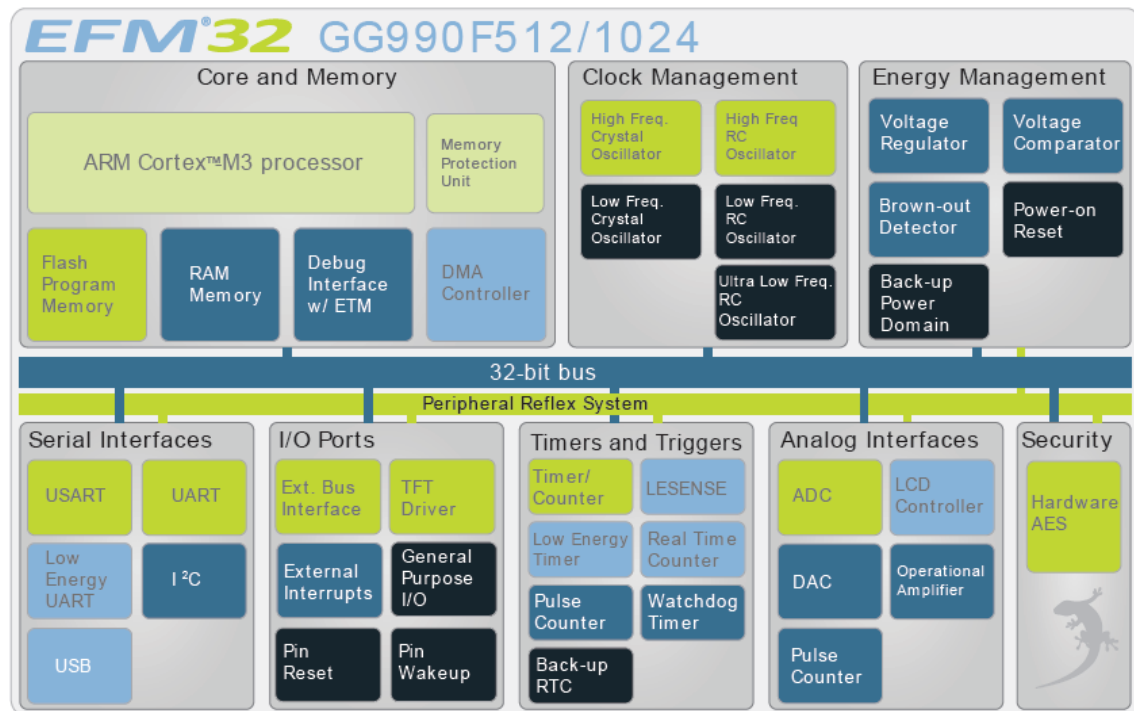


Figura 35 Diagrama de Blocos do microcontrolador

6.5.5. ECRÃ

A escolha do ecrã teve em conta vários fatores, como o tamanho reduzido para o género de dispositivo, o baixo consumo de energia e a capacidade de poder visualizar o seu conteúdo em diferentes ambientes. Para tal o escolhido foi o Sharp LS013B7DH03, porque contém as seguintes características:

- Painel Transflectivo com propriedades transmissivas para ambientes interiores e refletivas para ambientes exteriores.
- Modo Transmissivo melhorado com luz de fundo
- Tamanho do ecrã de 1.28", de forma a se adaptar facilmente ao pulso.



Figura 36 Aspeto do ecrã de 1.28"

6.5.6. BATERIA

Devido às restrições de espaço selecionou-se uma bateria de polímeros de lítio, uma vez que estas são maleáveis, o que permite que se adaptem ao espaço envolvente. Para além disso, estas têm uma grande densidade de potência, que promove mais vida útil, havendo ainda a vantagem da possibilidade de carregamentos sempre que necessário, sendo dispensável realizar um ciclo de carga e descarga. As principais características deste tipo de baterias são:

- Auto descarregamento muito baixo, comparativamente as baterias de níquel.
- Pouca manutenção, não é necessário fazer descargas periódicas porque a bateria não tem memória.
- Necessário um circuito de proteção para manter o nível de tensão dentro dos limites.
- Mais caras aproximadamente 40% em relação as baterias de níquel.

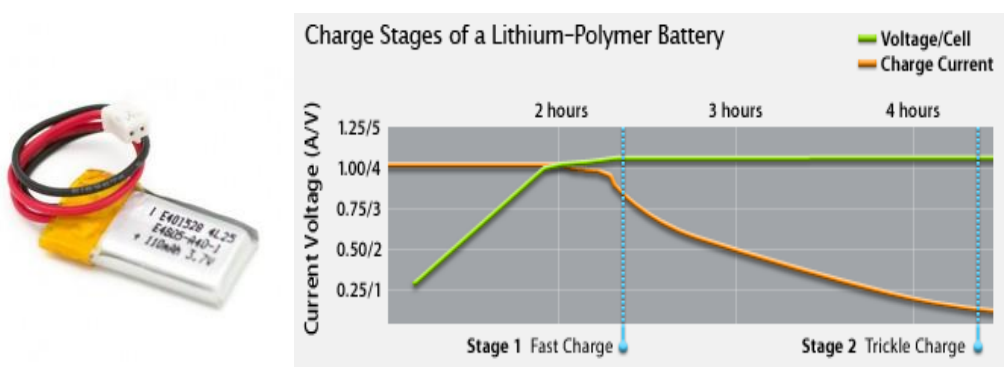


Figura 37 Estados de carga de uma bateria de polímeros de lítio

6.5.7. SENSOR DE TEMPERATURA

O sensor de temperatura para um equipamento *wearable* deve ser o mais pequeno possível. Para tal, o sensor escolhido (TMP006) tem um formato muito pequeno, 1,7x1,3 [mm], e mede a temperatura através da radiação infravermelha emitida pelos objetos. Através deste método o sensor pode ser colocado dentro do equipamento e evitar a necessidade de existirem aberturas no invólucro do dispositivo.

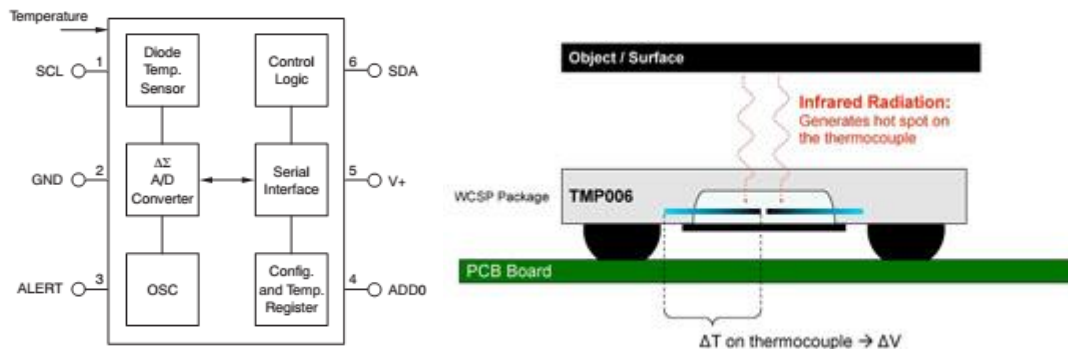


Figura 38 Sensor de temperatura

6.5.8. RESISTÊNCIA DA PELE

A medição da resistência da pele tem por objetivo determinar a humidade a que esta se encontra. Existe uma relação de proporcionalidade entre estas, pois à medida que uma aumenta a outra diminui. A solução encontrada para fazer esta medição passa por instalar dois pontos de contato com a pele, que por sua vez são medidos pelas entradas analógicas do microcontrolador.

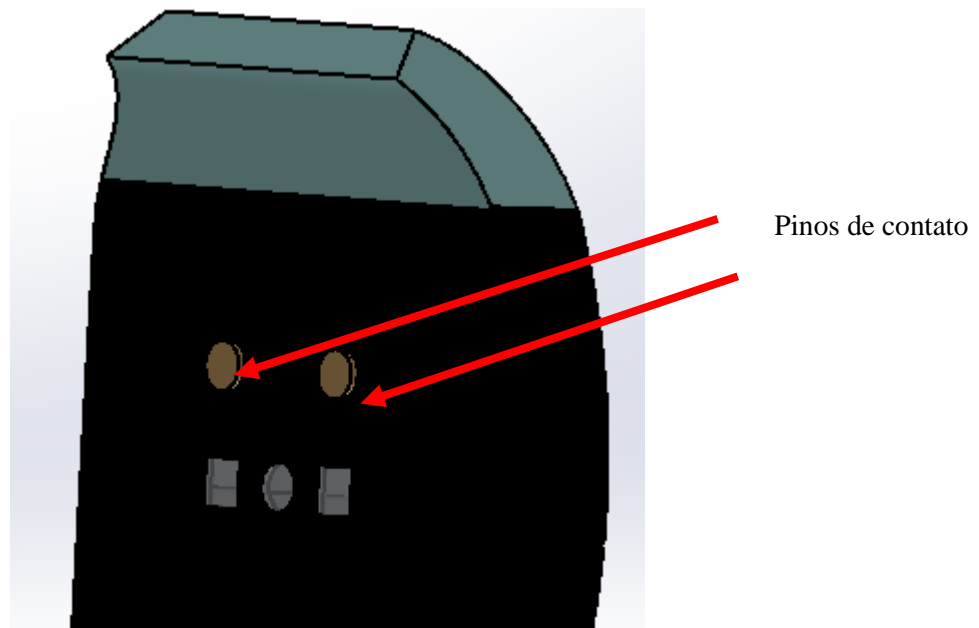


Figura 39 Pinos para medir a resistência da pele

6.5.9. BATIMENTO CARDÍACO

O batimento cardíaco é obtido através do sensor ótico TCRT1010. Este tem um formato compacto, e o emissor de luz está na mesma direção que o detetor, permitindo assim receber a luz refletida pelo objeto. O comprimento de onda é de 950 nm, o que corresponde a gama de luz infravermelha. A detecção é feita por um foto transístor.

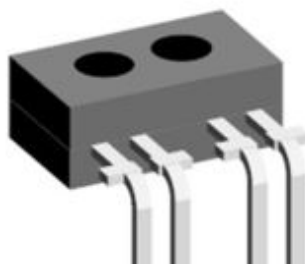


Figura 40 Aspeto do sensor ótico

7. ARQUITETURA DE SOFTWARE

7.1. ARQUITETURA GERAL

A arquitetura de *software* segue uma abordagem orientada a eventos, recorrendo ainda frequentemente ao esquema de publisher/subscriber, um conhecido paradigma para o desenvolvimento de sistemas distribuídos e caracterizado pelo desacoplamento das partes e pela comunicação assíncrona entre componentes.

A Figura 41 mostra os diversos componentes da arquitetura, as suas *interfaces* e o agrupamento nas diversas camadas existentes.

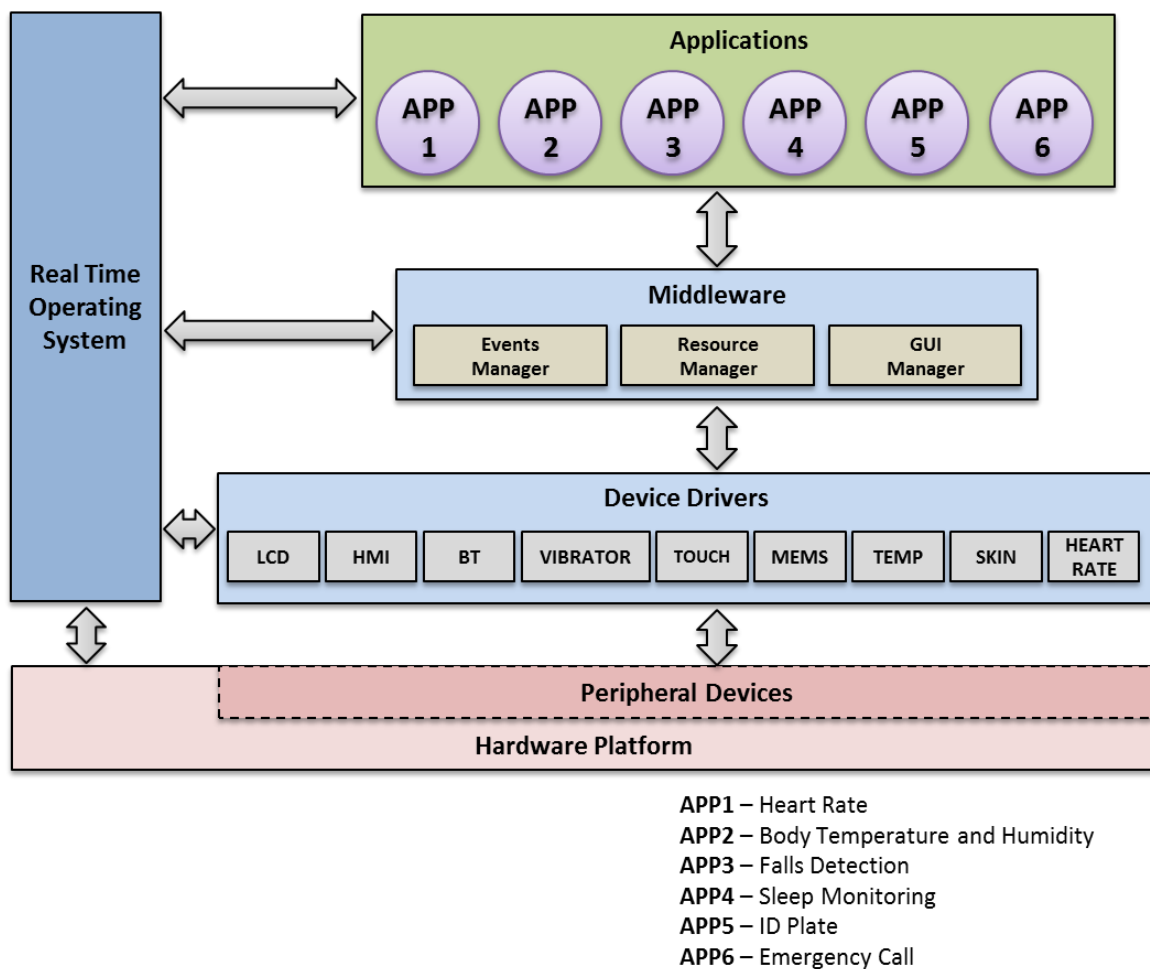


Figura 41 Arquitetura geral de *software*

7.2. DEVICE DRIVERS

Esta camada é constituída por um conjunto de drivers que permitem a interação do restante *software* com os dispositivos de *hardware* de uma forma simples e abstrata, ao invés de um nível demasiado detalhado.

7.2.1. LCD

A forma como a matriz de pixéis deste ecrã é utilizada para armazenar os dados da imagem deriva do seu nome (Sharp's Memory LCD). Cada pixel da matriz corresponde a 1 bit (apenas para escrita), correspondendo no total a um único buffer de gravação para uma imagem completa. Este recurso alivia o processador e barramento da sobrecarga de dados contínuos transferidos para atualizar a imagem, tendo em conta que a imagem só precisa de ser escrita uma vez (a imagem é retida indefinidamente). Quando é necessário atualizar a

imagem exibida, não há necessidade de limpar e reescrever a imagem inteira, basta apenas reescrever as linhas que se pretende alterar.

Os dados são enviados para o ecrã no formato little-endian. O tamanho da trama de dados por linha a escrever depende da resolução do ecrã. Neste caso específico a resolução do ecrã é de 128x128 o que representa um tamanho de 128 bits.

Inicialização dos recursos necessários para interagir com o ecrã:

```
EMSTATUS MEMLCD_Init(MEMLCD_Config *cfg)
{
    /* Setup clocks */
    CMU_ClockEnable(cmuClock_GPIO, true );
    CMU_ClockEnable(cfg->timerClock, true );
    CMU_ClockEnable(cfg->usartClock, true );
    /* Setup GPIO's */
    GPIO_PinModeSet(cfg->sclk.port, cfg->sclk.pin, gpioModePushPull, 0 );
    GPIO_PinModeSet(cfg->si.port,   cfg->si.pin,   gpioModePushPull, 0 );
    GPIO_PinModeSet(cfg->scs.port,   cfg->scs.pin,   gpioModePushPull, 0 );
    GPIO_PinModeSet(cfg->extcomin.port, cfg->extcomin.pin, gpioModePushPull, 0 );
    GPIO_PinModeSet(cfg->extmode.port, cfg->extmode.pin, gpioModePushPull, 0 );
    GPIO_PinModeSet(cfg->disp.port,   cfg->disp.pin,   gpioModePushPull, 0 );
}
```

Existem três comandos que podem ser enviados para o ecrã que foram implementados no driver:

- Apagar o ecrã (Clear Screen)
- Escrever uma linha (Write Line)
- Escrever múltiplas linhas (Write Multiple Lines)

Comando: Apagar Ecrã

Este comando limpa todo o ecrã, escrevendo 0 em todas as posições da memória. A estrutura de comando a tem o formato apresentado na figura seguinte.

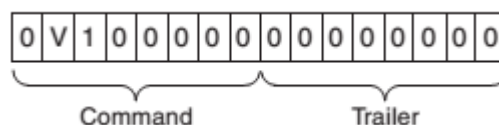


Figura 42 Estrutura de comando para apagar o ecrã

Implementação do comando apagar ecrã:

```
void MEMLCD_Clear( void )
{
    uint8_t cmd;
    /* Set SCS */
    GPIO_PinOutSet( pConfig->scs.port, pConfig->scs.pin );
    /* SCS setup time: min 6us */
    usDelay(6);
    /* Send command */
    cmd = (MEMLCD_CMD_ALL_CLEAR | comPolarity);
    USART_TxDouble( pConfig->usart, cmd );
    /* Wait for transfer to finish */
    while ( !(pConfig->usart->STATUS & USART_STATUS_TXC) );
    /* Clear SCS */
    GPIO_PinOutClear( pConfig->scs.port, pConfig->scs.pin );
}
```

Comandos: Escrever uma linha e Escrever múltiplas linhas

A estrutura do comando para escrever uma linha é a representada na figura seguinte.

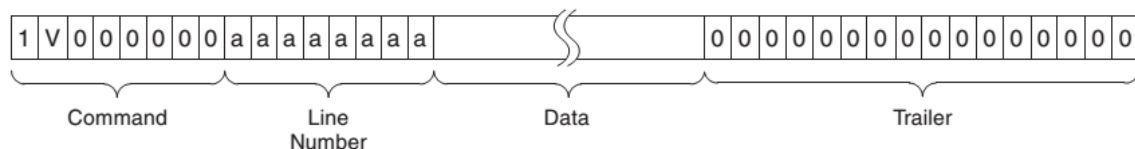


Figura 43 Estrutura de comando para escrever uma linha

O comando para escrever múltiplas linhas inicia-se da mesma forma do anterior utilizando a mesma estrutura inicial. No entanto após os 8 bits de dados devem ser enviados 8 bits de trailer (ao invés de 16bits) e repetir esta sequência para todas as linhas que se queria atualizar. No envio dos dados da última linha deve ser enviado o trailer com 16 bits para indicar o fim da transmissão.

Tendo em conta a similaridade dos comandos de escrita, foi criada uma função que tanto permite a escrita de apenas uma linha como de várias. A implementação pode ser vista no código apresentado abaixo:

```
EMSTATUS MEMLCD_Update(uint16_t *data, int firstLine, int lastLine)
{
    int i,j;
    /* Assert SCS */
    GPIO_PinOutSet( pConfig->scs.port, pConfig->scs.pin );
    /* SCS setup time: min 6us */
    usDelay(6);
    /* Send update command and first line address */
    USART_TxDouble(pConfig->usart, MEMLCD_CMD_UPDATE | (firstLine + 1) << 8);
    /* Get start address to draw from */
    uint16_t *p = (uint16_t *)data;
    p += firstLine * 10;
    for ( i=firstLine; i<=lastLine; i++ ) {
        /* Send pixels for this line */
        for ( j=0; j<9; j++ ) {
            USART_TxDouble(pConfig->usart, *p);
        }
    }
}
```



```

        p++;
    }
    /* Skip padding data in frame buffer */
    p += 1;
}
/* Wait for USART to finish */
while (!(pConfig->usart->STATUS & USART_STATUS_TXC)) ;
/* De-assert SCS */
GPIO_PinOutClear( pConfig->scs.port, pConfig->scs.pin );
return MEMLCD_OK;
}

```

7.2.2. BATIMENTO CARDÍACO

A medição do batimento cardíaco é feita através da variação do volume de sangue, uma vez que os dois são síncronos. Para este método é necessário termos uma fonte de luz que emita sobre os tecidos da pele, e um detetor de luz que meça a quantidade de luz que foi refletida. Esta quantidade de luz medida varia conforme o volume de sangue, e pode ser traduzida no batimento cardíaco.

Para fazer esta medição é necessário usar uma saída digital, para ativação do LED, e uma entrada analógica para ler a quantidade de luz recebida. A função a seguir configura a entrada analógica do microcontrolador.

```

static void ADC_HR_Config(void)
{
    ADC_Init_TypeDef init = ADC_INIT_DEFAULT;
    ADC_InitSingle_TypeDef singleInit = ADC_INITSINGLE_DEFAULT;
    init.timebase = ADC_TimebaseCalc(0);
    init.prescale = ADC_PrescaleCalc(7000000, 0);
    ADC_Init(ADC0, &init);
    singleInit.reference = adcRef2V5;
    singleInit.input = adcSingleInpCh5;
    ADC_InitSingle(ADC0, &singleInit);
}

```

O fluxograma que se segue mostra os passos necessários para fazer a aquisição do total de luz recebida pelo foto transístor, sendo o resultado final uma amostra de dez medições.

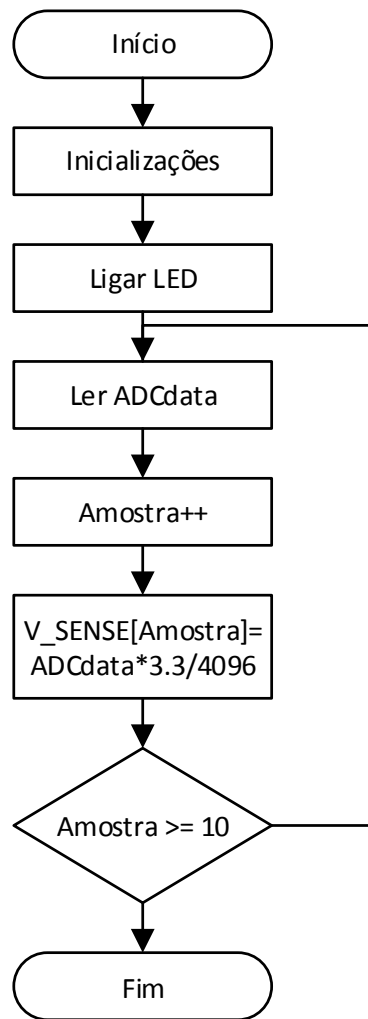


Figura 44 Fluxograma do driver de leitura do TCRT1010

7.2.3. RESISTÊNCIA DA PELE

Para fazer a medição da resistência da pele é necessário colocar uma tensão DC num contato que esteja a tocar na pele, e seguidamente ler qual a tensão presente noutro contato próximo. A diferença destas duas tensões irá resultar no valor da resistência da pele, se a tensão lida for igual à tensão de referência, quer dizer que a resistência da pele é muito baixa, se o valor lido for muito baixo, a resistência será mais alta. O fluxograma que se segue mostra os passos necessários para a determinação deste parâmetro.

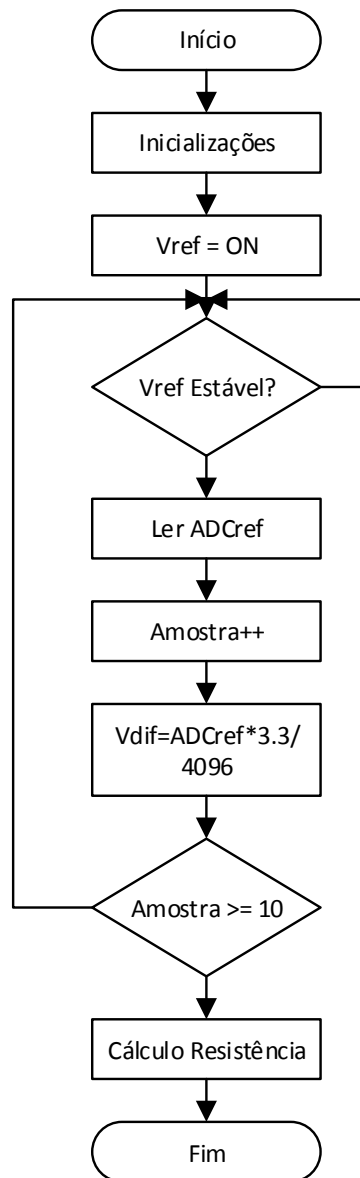


Figura 45 Fluxograma do driver de leitura da resistência da pele

7.2.4. SENSOR DE TEMPERATURA

O sensor de temperatura TMP006 é o primeiro de uma série de sensores de temperatura que medem a temperatura de um objeto sem a necessidade de entrar em contato com o mesmo. Este utiliza um sensor para absorver a energia infravermelha emitida a partir do objeto a ser medido.

A implementação do driver está representada no fluxograma da Figura 46. Este driver recorre ao protocolo I2C para comunicar com o sensor e posteriormente calcula tanto a temperatura ambiente como a do objeto.

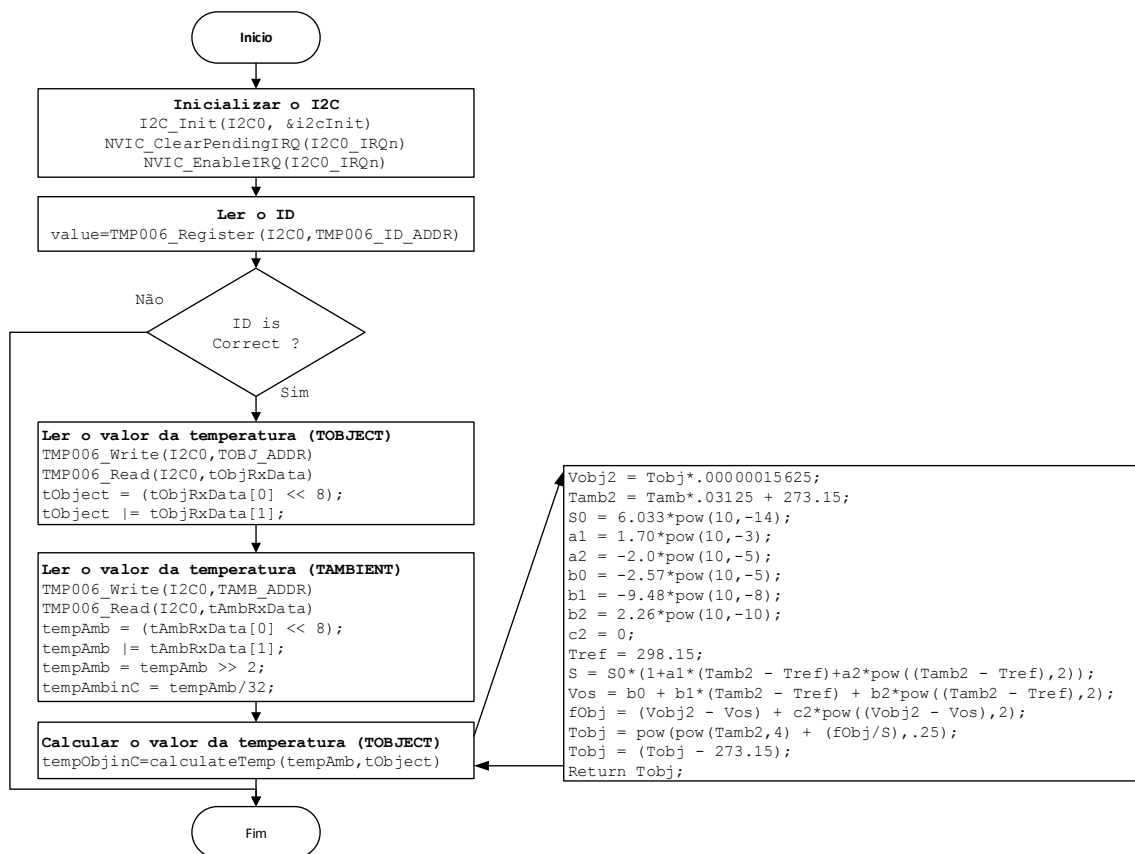


Figura 46 Fluxograma do driver do sensor TMP006

7.2.5. TOUCH

A detecção de contacto capacitiva é uma tecnologia baseada na medição de uma alteração na capacidade. O driver desenvolvido utiliza a interface de baixo consumo LESENSE que se baseia nos comparadores analógicos do microcontrolador EFM32 para medir a capacidade entre o pino sensor e a referência. Mais especificamente, é medida a capacidade entre o pino sensor e a referência através de um circuito oscilador RC (este circuito é intrínseco ao microcontrolador). Desta forma a frequência vai mudar em função da capacidade presente no pino sensor, como ilustrado na figura seguinte.

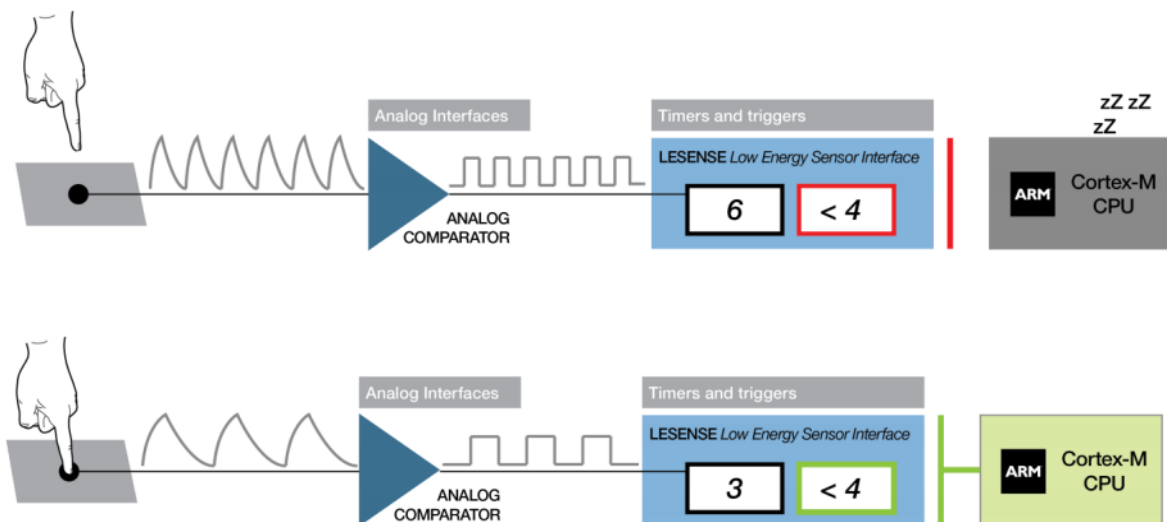


Figura 47 Comportamento do sensor capacitivo

Para uma correta aquisição dos segmentos capacitivos foi necessário implementar a seguinte sequencia:

- Configuração dos comparadores analógicos para o modo de toque capacitivo.
- A saída do comparador analógico é encaminhada para o registo de LESENSE que inicia a contagem. O contador do tempo de amostragem é iniciado ao mesmo tempo.
- Quando o contador de tempo de amostragem atinge o valor configurado, o valor de contagem do LESENSE é armazenado e comparado com o valor limite (*threshold*).
- Se a comparação do valor de LESENSE for superior ao valor limite, é despoletada uma interrupção, e posteriormente processada a informação relevante (qual o valor presente em cada segmento). De notar que o microcontrolador nunca é acordado durante um ciclo de medição, ajudando deste modo na poupança de energia.

Tendo em conta as características intrínsecas do microcontrolador, o circuito RC apenas precisa de ser associado aos pinos que se pretendem utilizar para os segmentos capacitivos (neste caso concreto são 4 segmentos) como ilustrado no seguinte bloco de código.

```
void CAPLESENSE_setup(void)
{
    /* Configure ACMP locations, ACMP output to pin disabled. */
    ACMP_GPIOSetup(ACMP1, 0, false, false);
    /* Initialize ACMPs in capacitive sense mode. */
    ACMP_CapsenseInit(ACMP1, &initACMP);
    /* Configure the drive strength of the ports for the light sensor. */
    GPIO_DriveModeSet(CAPLESENSE_SLIDER_PORT0, gpioDriveModeStandard);
    /* Initialize the 4 GPIO pins of the touch slider for using them as LESENSE
     * scan channels for capacitive sensing. */
    GPIO_PinModeSet(CAPLESENSE_SLIDER_PORT0, CAPLESENSE_SLIDER0_PIN,
        pioModeDisabled, 0);
}
```

```
    GPIO_PinModeSet(CAPLESENSE_SLIDER_PORT0,CAPLESENSE_SLIDER1_PIN,  
gpioModeDisabled,0);  
    GPIO_PinModeSet(CAPLESENSE_SLIDER_PORT0,CAPLESENSE_SLIDER2_PIN,  
gpioModeDisabled,0);  
    GPIO_PinModeSet(CAPLESENSE_SLIDER_PORT0,CAPLESENSE_SLIDER3_PIN,  
gpioModeDisabled,0);  
}
```

Depois de configurados todos os parâmetros relativos à aquisição capacitiva (pinos e valores limite para comparação), é apenas necessário garantir que na rotina de interrupção (despoletada apenas quando pelo menos um dos segmentos é superior ao limite) são lidos e guardados para processamento todos os valores.

O fluxograma da Figura 48 ilustra a implementação desta rotina.

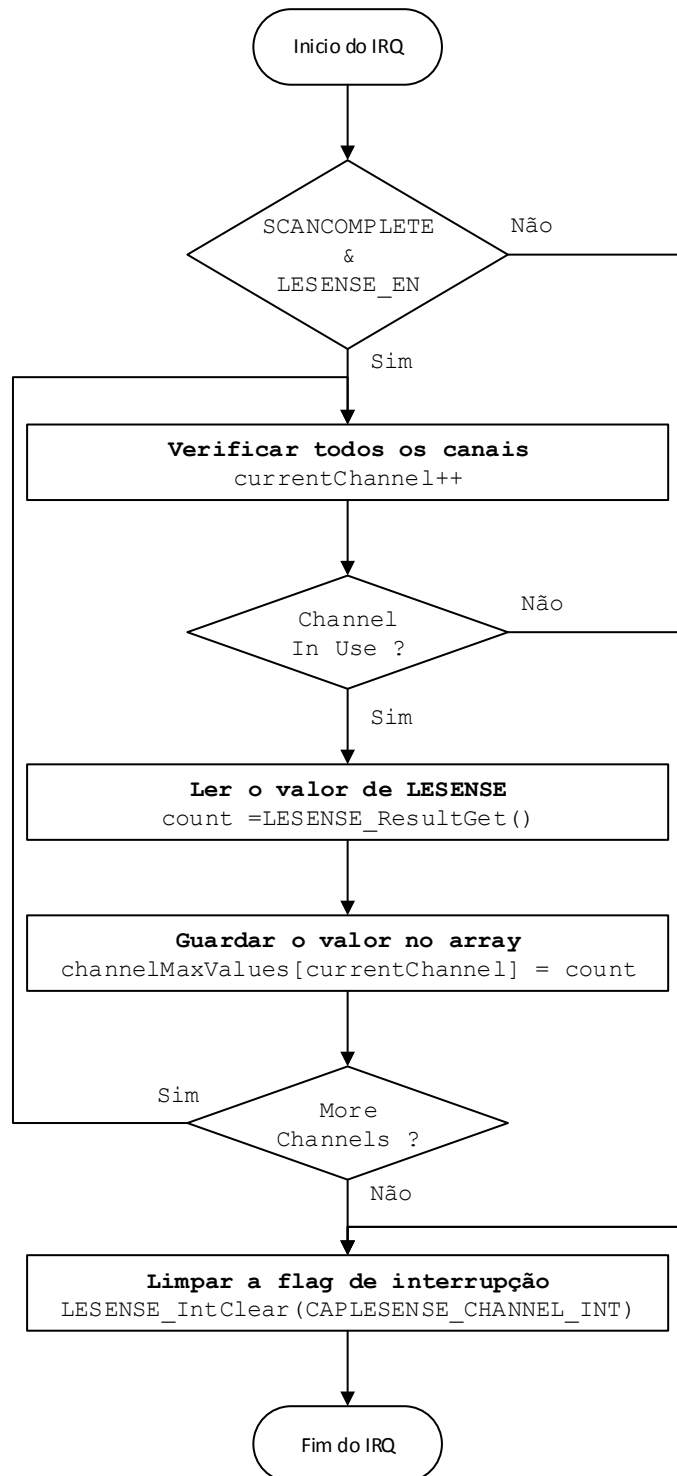


Figura 48 Fluxograma da rotina de interrupção do TOUCH

7.2.6. MEMS

Uma das características mais interessantes do sensor escolhido (LIS3DSH) é a possibilidade da configuração de máquinas de estados independentes da intervenção do microprocessador. Este sensor possui duas máquinas de estados, cada uma com a possibilidade de configuração de 16 estados, tendo como parâmetros:

- 4 Temporizadores independentes
- 2 Mascaras independentes (X, Y, Z, V)
- 3 Limites de aceleração independentes
- Função de detecção de picos

O próprio sensor já contém algumas definições para eventos específicos como:

- Clique (Toggle)
- Duplo clique (Double Tap)
- Acordar do estado parado (Wake Up)

A capacidade de acordar do estado parado, é bastante interessante tendo em conta o objetivo de poupança de energia. Com este evento o microprocessador apenas necessita de realizar as tarefas cíclicas se o dispositivo não estiver parado.

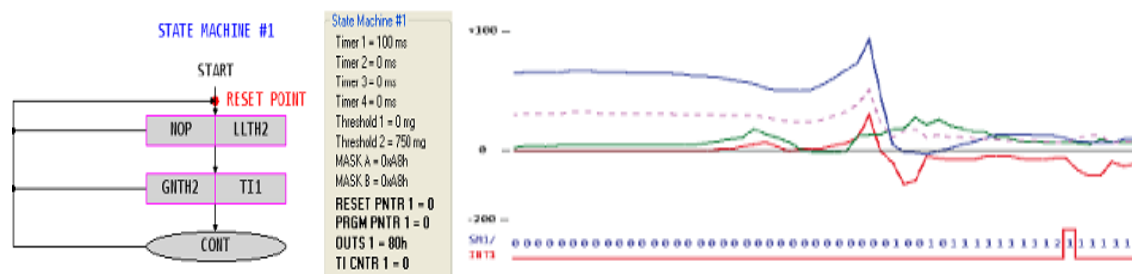


Figura 49 Exemplo da máquina de estados para a função acordar

O fluxograma da Figura 50 ilustra o driver implementado tendo em conta o protocolo específico do LIS3DSH.

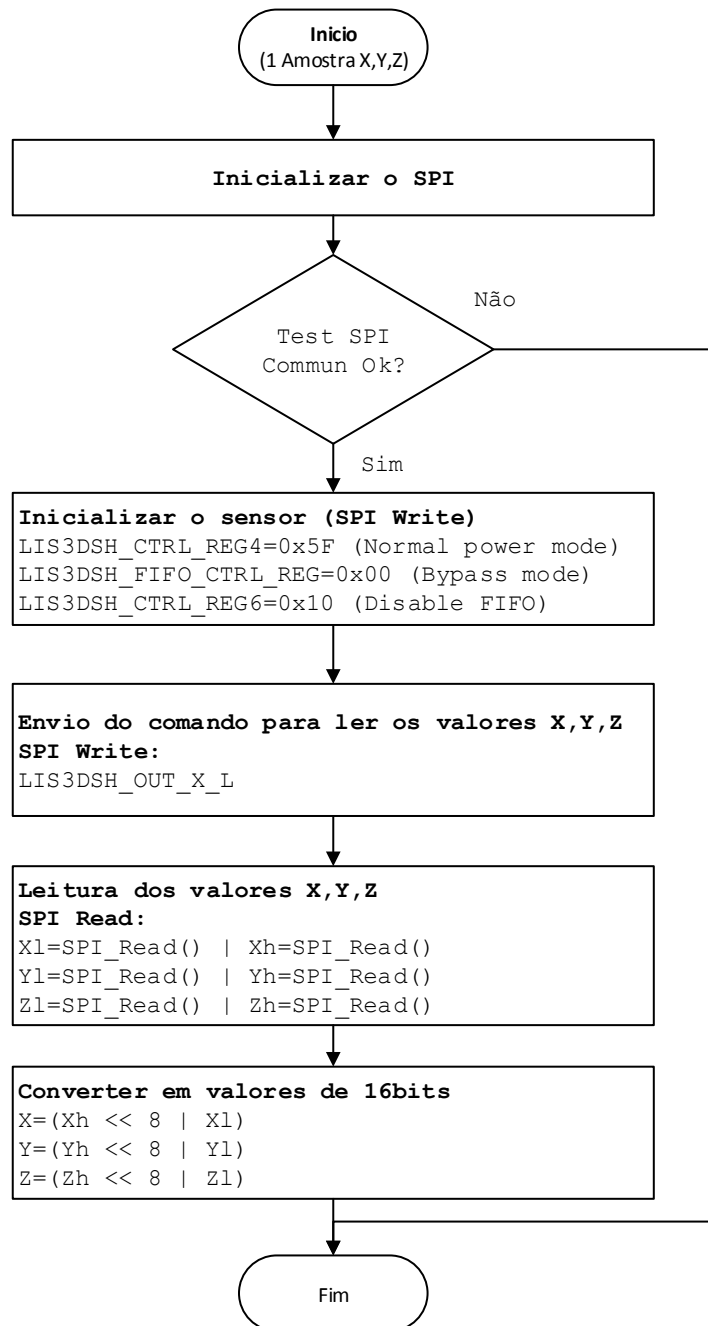


Figura 50 Fluxograma da comunicação com o MEMS

7.3. SISTEMA OPERATIVO

Esta camada é onde estão contidas todas as rotinas e funcionalidades pertencentes ao funcionamento do sistema operativo.

O RTX implementa a interface genérica de RTOS para processadores Cortex-M através da API CMSIS-RTOS. Esta API disponibiliza uma interface uniforme para qualquer componente de software que necessite de implementar funcionalidades de RTOS.

Cada aplicação que é criada utilizando o RTX tem de incluir o ficheiro “cmsis-os.h” que contém a API de interface ao RTX CMSIS-RTOS. A configuração do sistema operativo e de algumas das suas funcionalidades encontram-se no ficheiro “RTX_Conf_CM.c”.

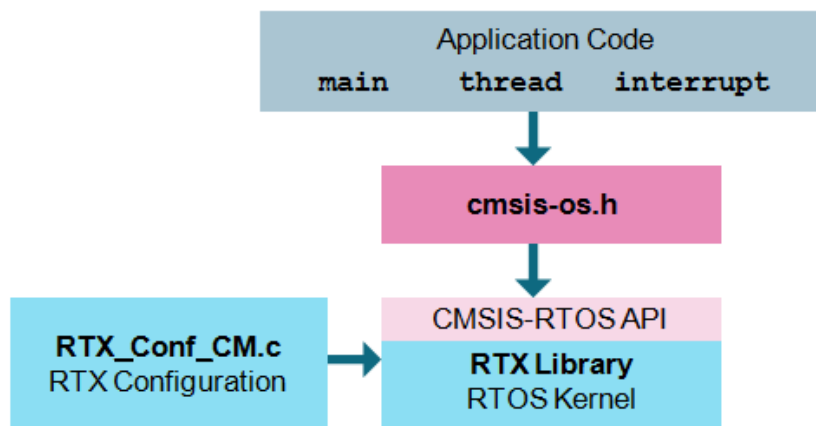


Figura 51 Estrutura do CMSIS-RTOS RTX

O RTX disponibiliza um conjunto de componentes de *software* que podem ser utilizados para a criação das várias aplicações constituintes do sistema. Este tipo de componentes está dividido nos seguintes grupos:

- Thread
- Mutex
- Semaphore
- Message queue
- Memory pool
- Mail Queue
- Timer

7.3.1. THREAD

Este conjunto de funções permite criar, definir e controlar funções de *thread*. Um exemplo de uma *thread*, é a função *main* que é iniciada quando o sistema é iniciado.

Uma thread pode ter os seguintes estados:

- **Running** – A *thread* está ativa e está a correr, sendo que apenas uma *thread* pode estar ativa de cada vez.
- **Ready** – Após uma *thread* ter terminado a sua execução ou passar para o estado *Waiting*, a thread que tem maior prioridade que se encontra no estado *Ready* passa a ficar ativa.
- **Waiting** – As *threads* que se encontram neste estado são *threads* que estão à espera que ocorra um evento.
- **Inactive** – *Threads* que ainda não foram criadas ou que foram terminadas, passam para este estado.

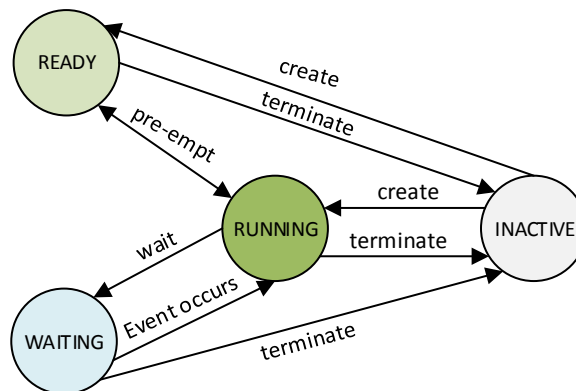


Figura 52 Estados das *threads*

Exemplo da utilização de *threads*:

```
#include "cmsis_os.h"

DigitalOut led1(LED1);
DigitalOut led2(LED2);

void led2_thread(void const *args) {
    while (true) {
        led2 = !led2;
        osDelay(1000);
    }
}

osThreadDef(led2_thread, osPriorityNormal, DEFAULT_STACK_SIZE);

int main() {
    osThreadCreate(osThread(led2_thread), NULL);
}
```

```

while (true) {
    led1 = !led1;
    osDelay(500);
}

```

7.3.2. MUTEX

As funções do grupo *Mutex* são utilizadas para sincronizar a execução de *threads*. Podem ser utilizadas para a proteção do acesso a recursos compartilhados entre duas *threads*.

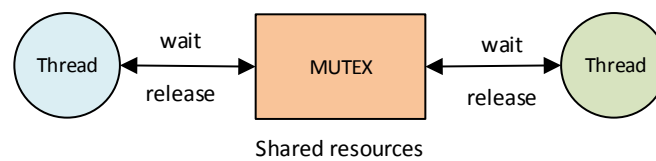


Figura 53 Sequência de operação de um *Mutex*

Exemplo da utilização de *mutex*:

```

#include "cmsis_os.h"

osMutexId stdio_mutex;
osMutexDef(stdio_mutex);

void notify(const char* name, int state) {
    osMutexWait(stdio_mutex, osWaitForever);
    printf("%s: %d\n\r", name, state);
    osMutexRelease(stdio_mutex);
}

void test_thread(void const *args) {
    while (true) {
        notify((const char*)args, 0); osDelay(1000);
        notify((const char*)args, 1); osDelay(1000);
    }
}

void t2(void const *argument) {test_thread("Th 2");}
osThreadDef(t2, osPriorityNormal, DEFAULT_STACK_SIZE);

void t3(void const *argument) {test_thread("Th 3");}
osThreadDef(t3, osPriorityNormal, DEFAULT_STACK_SIZE);

int main() {
    stdio_mutex = osMutexCreate(osMutex(stdio_mutex));

    osThreadCreate(osThread(t2), NULL);
    osThreadCreate(osThread(t3), NULL);

    test_thread((void *) "Th 1");
}

```

7.3.3. SEMAPHORE

O grupo de funções relacionadas com os semáforos é utilizado para a gestão e proteção de recursos partilhados. Um exemplo da sua utilização é a gestão de acesso a um grupo de periféricos, em que o nº de recursos disponíveis é especificado por um parâmetro na criação dos semáforos.

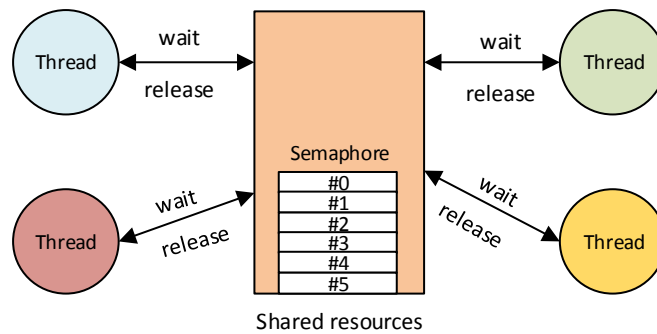


Figura 54 Operações dos semáforos

Exemplo da utilização de *semaphores*:

```
#include "cmsis_os.h"

osSemaphoreId two_slots;
osSemaphoreDef(two_slots);

void test_thread(void const *name) {
    while (true) {
        osSemaphoreWait(two_slots, osWaitForever);
        printf("%s\n\r", (const char*)name);
        osDelay(1000);
        osSemaphoreRelease(two_slots);
    }
}

void t2(void const *argument) {test_thread("Th 2");}
osThreadDef(t2, osPriorityNormal, DEFAULT_STACK_SIZE);

void t3(void const *argument) {test_thread("Th 3");}
osThreadDef(t3, osPriorityNormal, DEFAULT_STACK_SIZE);

int main (void) {
    two_slots = osSemaphoreCreate(osSemaphore(two_slots), 2);

    osThreadCreate(osThread(t2), NULL);
    osThreadCreate(osThread(t3), NULL);

    test_thread((void *) "Th 1");
}
```

7.3.4. MESSAGE QUEUE

As funções incluídas no grupo de *Message Queue* permitem criar, controlar, enviar, receber ou esperar por mensagens.

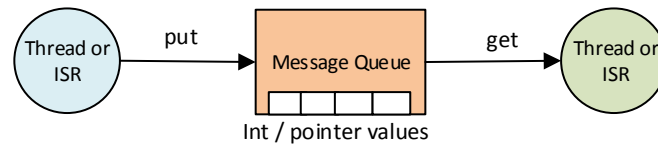


Figura 55 Estrutura de uma *Message Queue*

7.3.5. MEMORY POOL

O *memory pool* são partições de memória com um tamanho fixo. Ao contrário da memória dinâmica, a memória é sempre fixa trazendo benefícios em termos de tempos de execução, pois sabemos o tamanho de todos os blocos de memória.

7.3.6. MAIL QUEUE

O *Mail* (correio) é um bloco de memória que é enviado para uma *thread* ou rotina de interrupção. Este grupo de funções permite criar, controlar, enviar e receber correio.

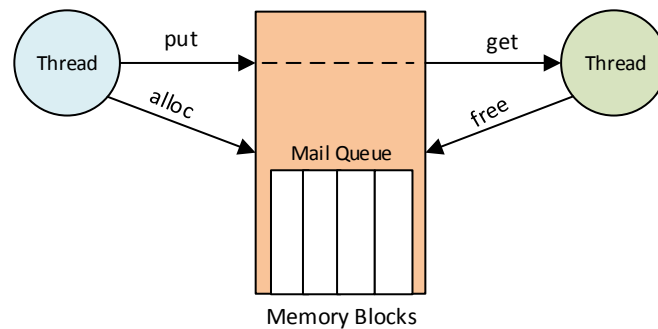


Figura 56 Estrutura de um *Mail Queue*

Exemplo da utilização de uma *Mail Queue*:

```
#include "cmsis_os.h"

typedef struct {
    float    voltage; /* AD result of measured voltage */
    float    current; /* AD result of measured current */
    uint32_t counter; /* A counter value */
} mail_t;

osMailQDef(mail_box, 16, mail_t);
osMailQId mail_box;
```

```

void send_thread (void const *args) {
    uint32_t i = 0;
    while (true) {
        i++; // fake data update
        mail_t *mail = (mail_t*)osMailAlloc(mail_box, osWaitForever);
        mail->voltage = (i * 0.1) * 33;
        mail->current = (i * 0.1) * 11;
        mail->counter = i;
        osMailPut(mail_box, mail);
        osDelay(1000);
    }
}

osThreadDef(send_thread, osPriorityNormal, DEFAULT_STACK_SIZE);

int main (void) {
    mail_box = osMailCreate(osMailQ(mail_box), NULL);
    osThreadCreate(osThread(send_thread), NULL);

    while (true) {
        osEvent evt = osMailGet(mail_box, osWaitForever);
        if (evt.status == osEventMail) {
            mail_t *mail = (mail_t*)evt.value.p;
            printf("\nVoltage: %.2f V\n\r", mail->voltage);
            printf("Current: %.2f A\n\r", mail->current);
            printf("Number of cycles: %u\n\r", mail->counter);

            osMailFree(mail_box, mail);
        }
    }
}

```

7.3.7. TIMER

A função timer é chamada quando um período de tempo expira. O timer pode ser iniciado, terminado e parado.

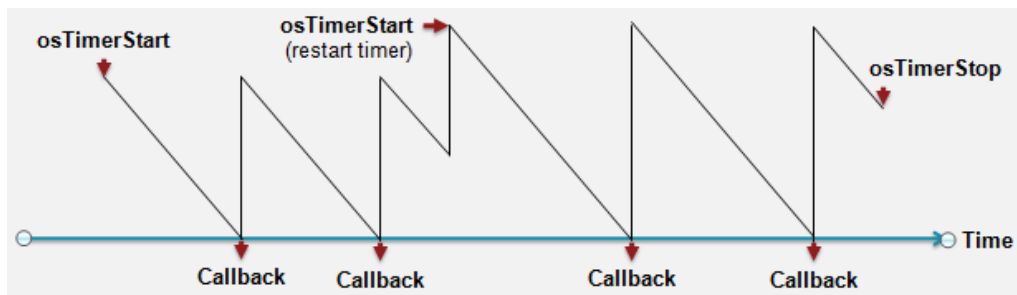


Figura 57 Sequência temporal de uma função *timer*

Exemplo da utilização do *timer*:

```

#include "cmsis_os.h"

DigitalOut LEDs[4] = {
    DigitalOut(LED1), DigitalOut(LED2), DigitalOut(LED3), DigitalOut(LED4)
};

void blink(void const *n) {
    LEDs[(int)n] = !LEDs[(int)n];
}

```

```

}

osTimerDef(blink_0, blink);
osTimerDef(blink_1, blink);
osTimerDef(blink_2, blink);
osTimerDef(blink_3, blink);

int main(void) {
    osTimerId timer_0 = osTimerCreate(osTimer(blink_0), osTimerPeriodic, (void
*)0);
    osTimerId timer_1 = osTimerCreate(osTimer(blink_1), osTimerPeriodic, (void
*)1);
    osTimerId timer_2 = osTimerCreate(osTimer(blink_2), osTimerPeriodic, (void
*)2);
    osTimerId timer_3 = osTimerCreate(osTimer(blink_3), osTimerPeriodic, (void
*)3);

    osTimerStart(timer_0, 2000);
    osTimerStart(timer_1, 1000);
    osTimerStart(timer_2, 500);
    osTimerStart(timer_3, 250);

    osDelay(osWaitForever);
}

```


7.4. MIDDLEWARE

A camada de *middleware* é responsável por gerir todos os acessos ao *hardware* (através dos drivers) de forma a evitar concorrência e prevenir uma sobrecarga de acessos a determinado recurso por parte das várias aplicações além de garantir uma interoperabilidade entre as várias camadas de software.

7.4.1. GESTOR DE RECURSOS (*RESOURCE MANAGER*)

O gestor de recursos é responsável pela abstração do *hardware* (drivers de sensores), garantindo sempre a acessibilidade aos diversos recursos pelas aplicações prioritárias, e baixando o consumo de energia desligando recursos não essenciais a aplicações que se encontram a correr.

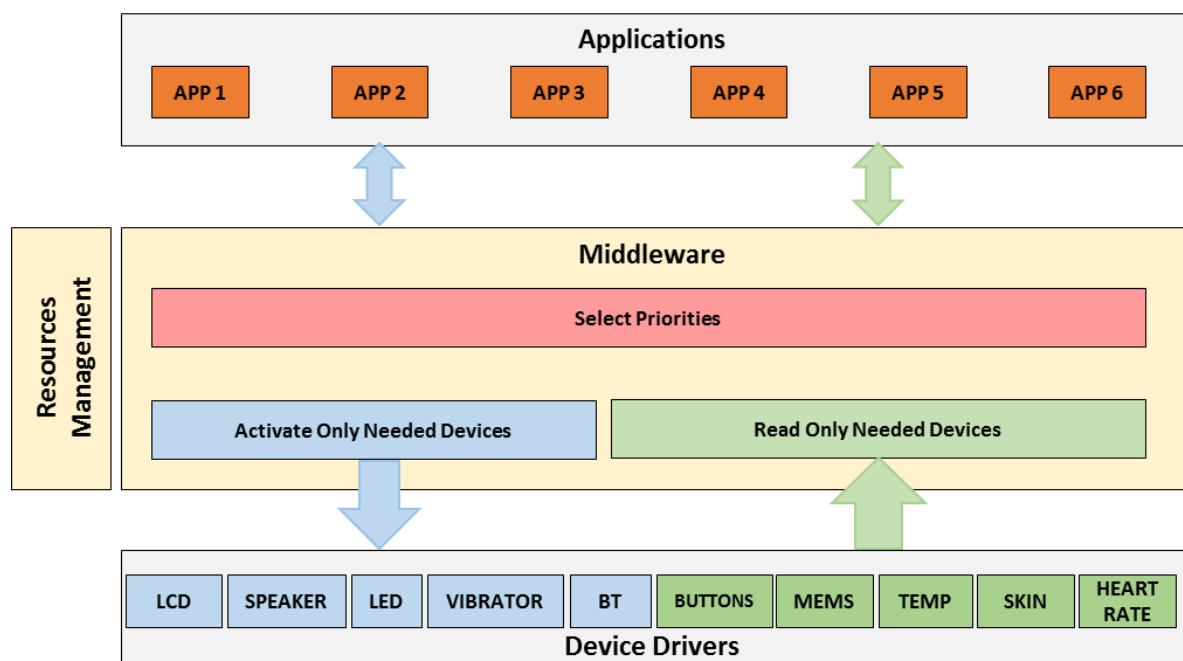


Figura 58 Gestor de Recursos (*Resources Management*)

De modo a garantir uma arquitetura flexível do ponto de vista das aplicações, este agente permite o seguinte:

- **Activate Only Needed Devices** – O gestor liga e desliga os componentes conforme a necessidade das aplicações, mantendo sempre ligados os essenciais para os serviços presentes.

- **Read Only Needed Devices** – Sempre que algum componente tente enviar alguma informação para uma aplicação, e esta interfira com o bom funcionamento de outra, esta tarefa gere a situação. Por exemplo, quando o utilizador está a fazer a monitorização do sono, e acidentalmente é pressionado o botão de retroceder, esta ação não irá fazer com que a aplicação feche. No caso de o botão de emergência ser pressionado, este irá se sobrepor a qualquer outro componente que tente comunicar.
- **Select Priorities** – Aqui é onde as aplicações subscvem aos recursos.

7.4.2. GESTOR DE INTERFACE DE UTILIZADOR (*USER INTERFACE MANAGEMENT*)

O gestor do interface gráfico é responsável pela gestão de todos os acessos ao ecrã, bem como pela transparência como são feitas as escritas no mesmo.

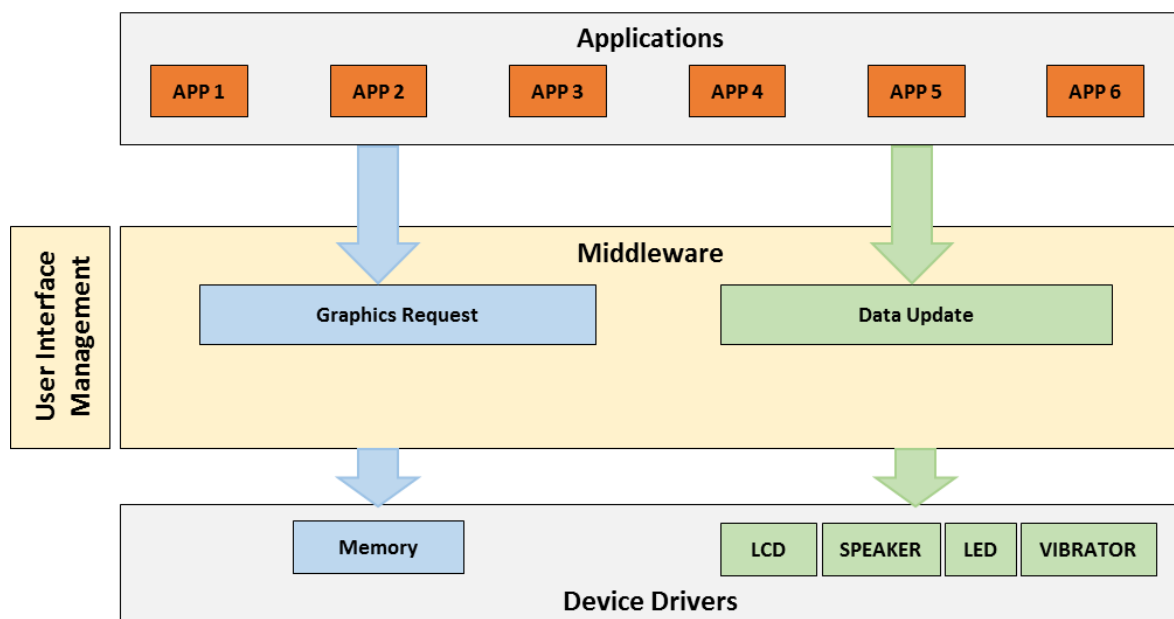


Figura 59 Gestor da Interface de utilizador (*User Interface Management*)

As aplicações que desejem escrever no ecrã necessitam apenas de aceder a dois recursos, que são:

- **Graphics Request** – Este recurso diz respeito ao modelo da imagem que é constituído por uma estrutura de pixéis completa (128x128). Desta forma o gestor apenas necessita de escrever diretamente todos os pixéis no ecrã (isto se a imagem for diferente da imagem atual presente no ecrã).

- **Data Update** - Sempre que for necessária a utilização de ecrãs dinâmicos, a aplicação deve utilizar este recurso, bastando para isso indicar qual a zona de pixéis a atualizar. Desta forma é reduzida a quantidade de informação a ser escrita no ecrã bem como a quantidade de recursos transferidos pelo sistema.

7.4.3. GESTOR DE EVENTOS (*EVENT MANAGER*)

O gestor de eventos é responsável por filtrar todas as interrupções, de modo a servir a aplicação que tem maior prioridade. Este é usado como interface entre a camada aplicacional e os *devices drivers*, de modo a focar a aplicação no seu único propósito.

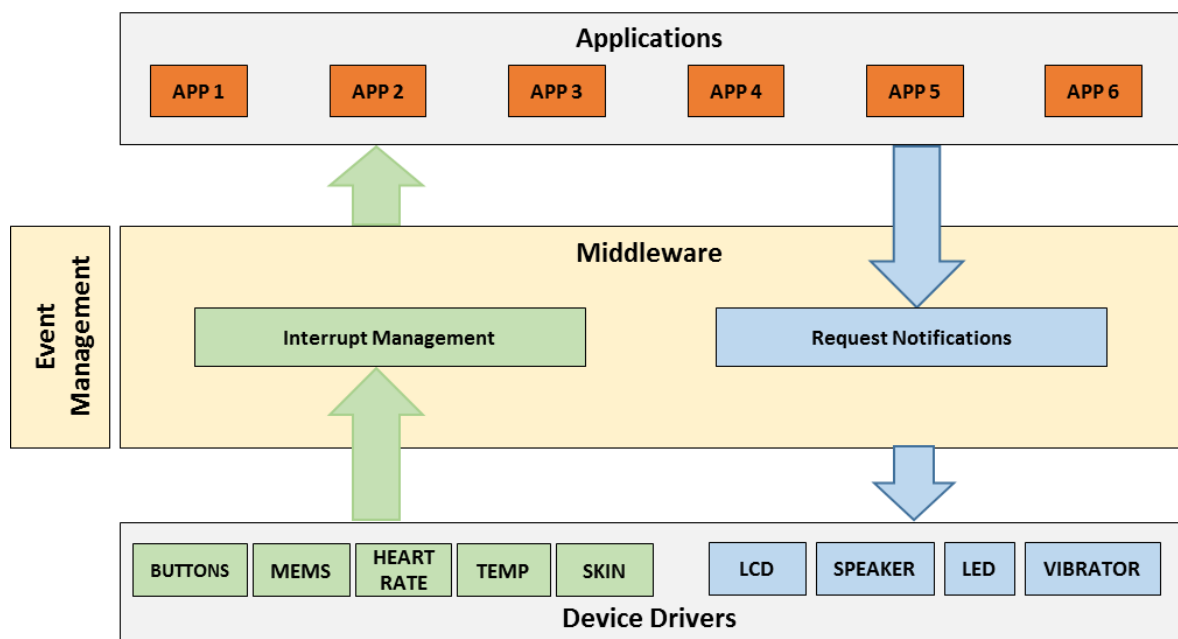


Figura 60 Gestor de Eventos (*Event Management*)

A implementação deste gestor tem dois blocos essenciais ao seu funcionamento:

- **Interrupt Manager** – Esta tarefa subscreve e gere todas as interrupções relacionadas com o *hardware* de entrada, como os botões de pressão e de toque, e os eventos programados do acelerómetro, tais como a deteção de quedas. Por exemplo, se o utilizador está no modo de visualização do batimento cardíaco, as interrupções que venham do sensor de temperatura não têm relevância, logo o *Interrupt Manager* vai descartar as mesmas.

- ***Request Notifications*** – Esta função tem como principal objetivo transformar as diversas notificações enviadas pelas aplicações, em eventos que possam ser percebidos pelo utilizador. Por exemplo, se a aplicação de deteção de quedas se inicia, esta função irá receber ordem para ligar todas as notificações que correspondem á mesma.

7.5. APLICAÇÕES

De seguida são apresentadas as aplicações presentes no dispositivo,

7.5.1. BATIMENTO CARDÍACO

Sempre que o utilizador queira visualizar qual o seu batimento cardíaco tem que executar os seguintes passos, conforme exemplifica na Figura 61:

1. Escolher o menu de batimento cardíaco através dos botões de seleção tátil.
2. Pressionar o botão de seleção.
3. A visualização do batimento aparece no ecrã assim que estiver disponível

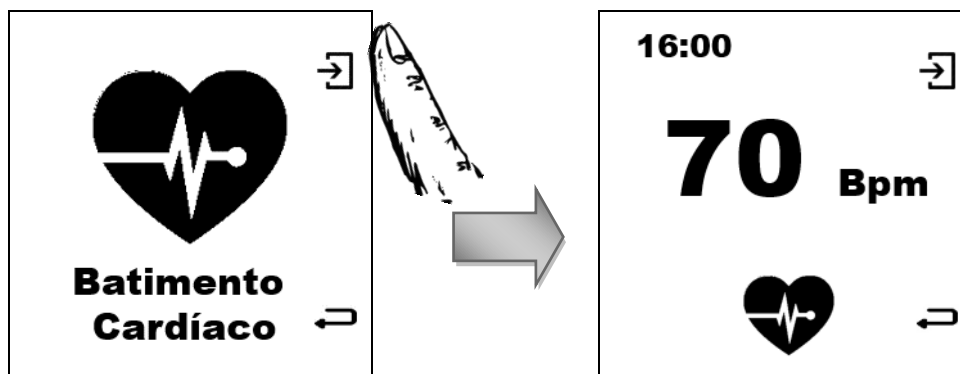


Figura 61 Menu de Batimento Cardíaco

De seguida seguem algumas imagens da implementação da aplicação, de forma a mostrar o efeito real.

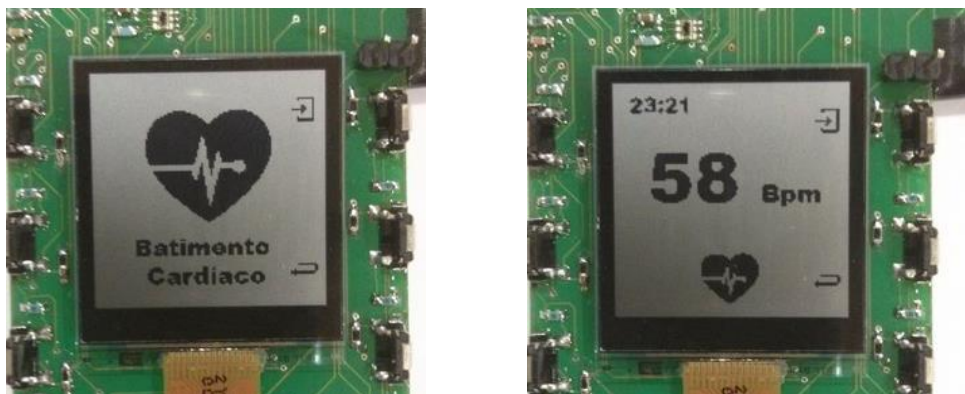


Figura 62 Implementação real da aplicação de batimento cardíaco

O fluxograma que se segue exemplifica a operação necessária para ler o batimento cardíaco. Após o utilizador estar no ecrã da opção que deseja, deve pressionar o botão de seleção para iniciar a aplicação, ou então se preferir pode sempre voltar ao menu imediatamente anterior pressionando o botão de retroceder. Caso o utilizador pressione o botão de seleção, é imediatamente ativado o sensor e apresentado no ecrã o respetivo valor medido. Para sair desta aplicação o botão de retroceder deve ser premido, o que levará a ser apresentado o ecrã de escolha de aplicações.

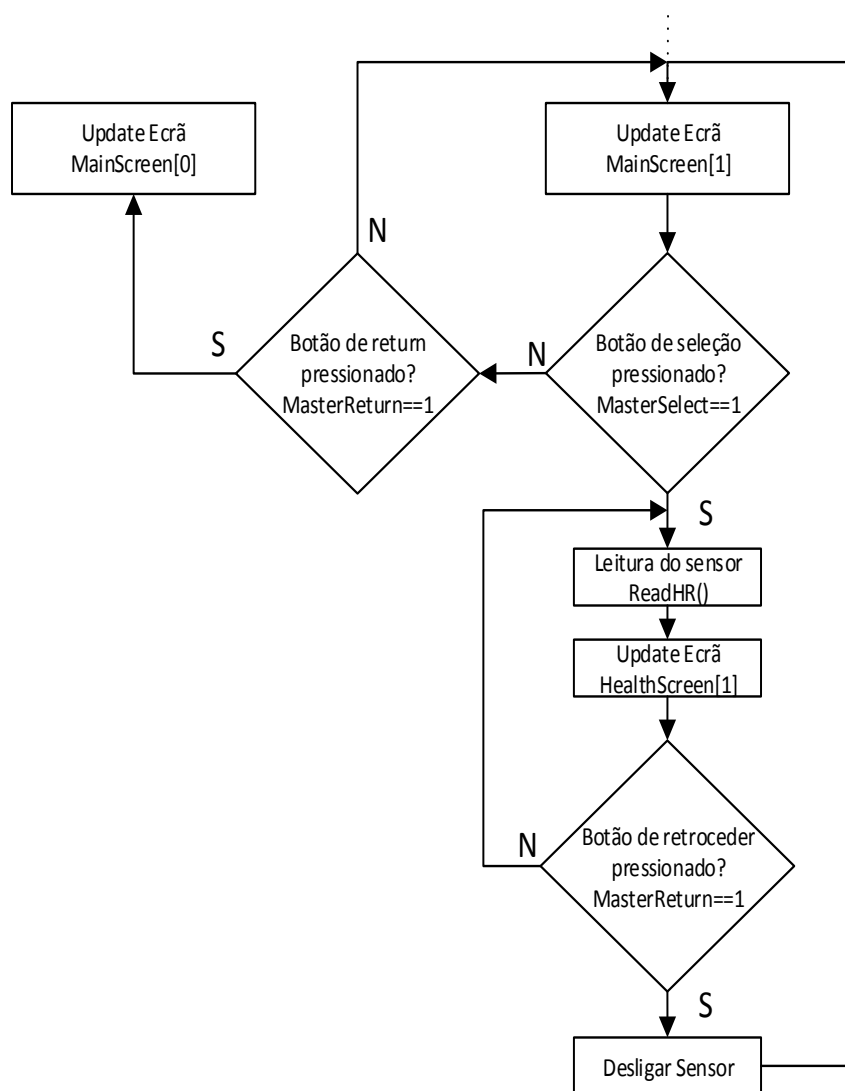


Figura 63 Fluxograma para aceder à aplicação do batimento cardíaco

7.5.2. TEMPERATURA E HUMIDADE CORPORAL

Sempre que o utilizador queira visualizar qual a sua temperatura corporal ou humidade tem que executar os seguintes passos:

1. Escolher o menu de temperatura através dos botões de seleção tátil.
2. Pressionar o botão de seleção.
3. A visualização da temperatura e humidade aparece no ecrã assim que estiver disponível

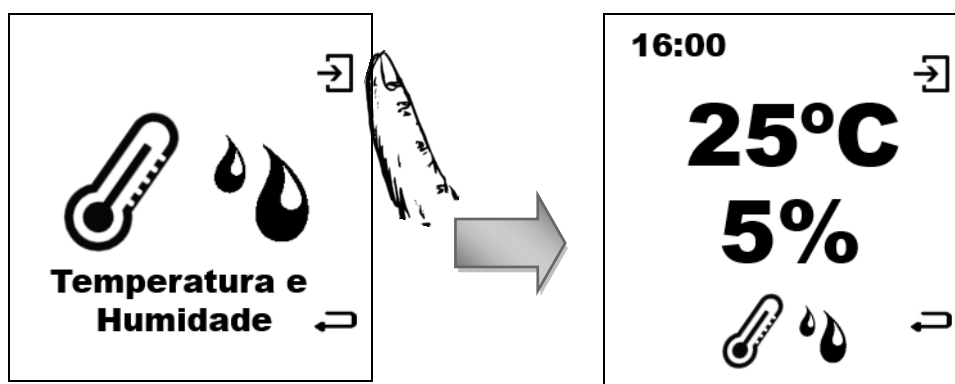


Figura 64 Menu de Temperatura e Humidade

Na imagem da Figura 65 pode-se ver o menu da aplicação.

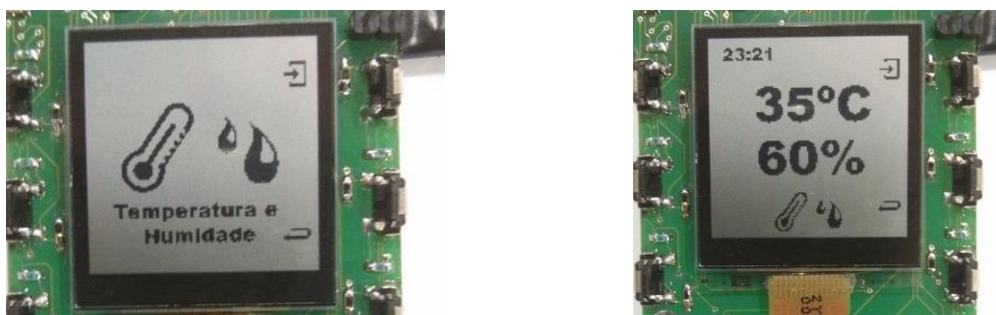


Figura 65 Implementação real da aplicação de Temperatura e Humidade

O fluxograma que se segue exemplifica os passos necessários para aceder á função de monitorização de temperatura e humidade. Assim como, voltar ao menu principal.

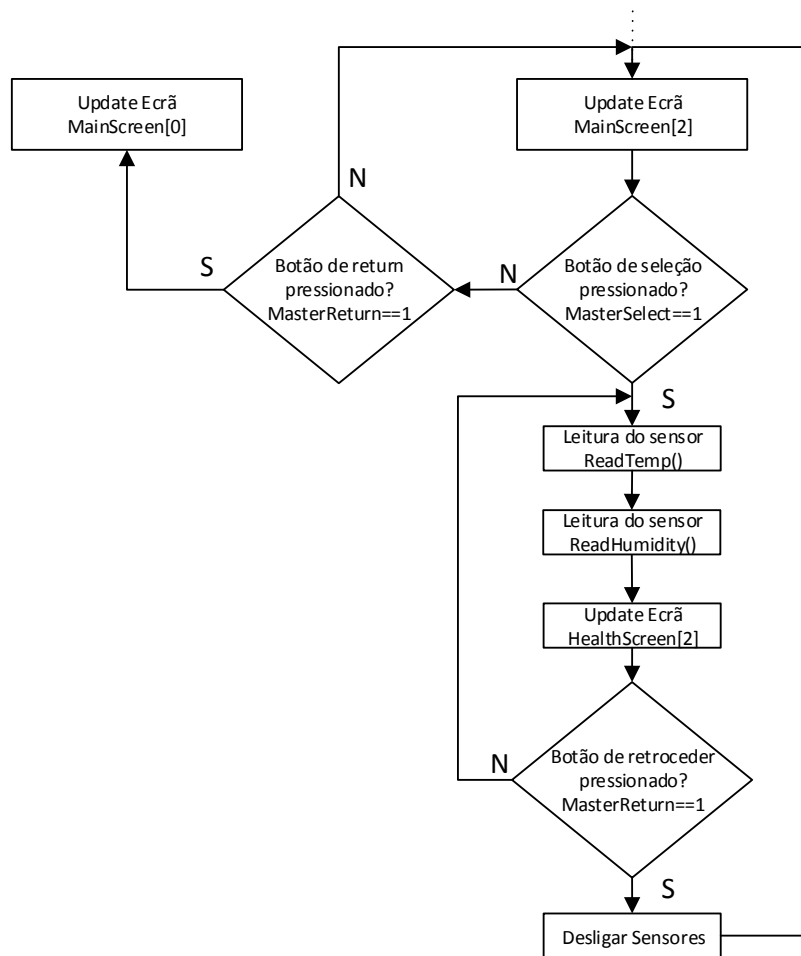


Figura 66 Fluxograma para aceder à aplicação de temperatura e humidade

7.5.3. DETEÇÃO DE QUEDAS

A deteção de quedas é um serviço que se encontra por defeito ativo no equipamento. Sempre que é detetada alguma queda, o equipamento entra em modo de alerta, que se caracteriza pelo seguinte:

1. Alertas intermitentes de vibração e som.
2. Luz do LED em vermelho e intermitente.
3. Mensagem no ecrã para confirmar ou não o evento.

4. Caso o utilizador não tenha nenhuma reação, o sistema inicia a chamada de emergência para o número previamente selecionado na aplicação do *smartphone*.

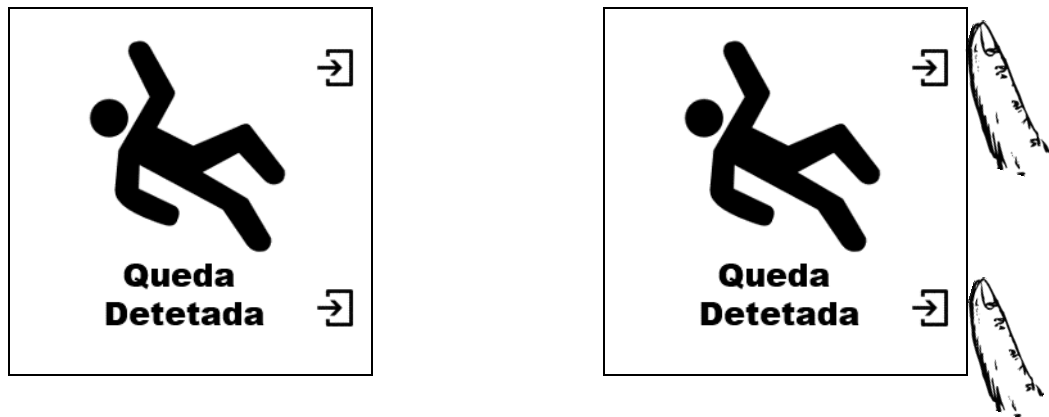


Figura 67 Menu de Detecção de Quedas

Quando uma queda é detetada, é enviada a imagem que se pode ver na para o ecrã. De modo a indicar o acontecimento.



Figura 68 Ecrã após deteção de queda

No fluxograma que se segue é explicado como o modo de deteção de quedas atua, e os procedimentos existentes para evitar chamadas de emergência desnecessárias. Se após 30 segundos de deteção de uma queda o utilizador não confirmar que se encontra consciente, pressionando ao mesmo tempo o botão de seleção e retroceder, o sistema inicia o modo de chamada de emergência.

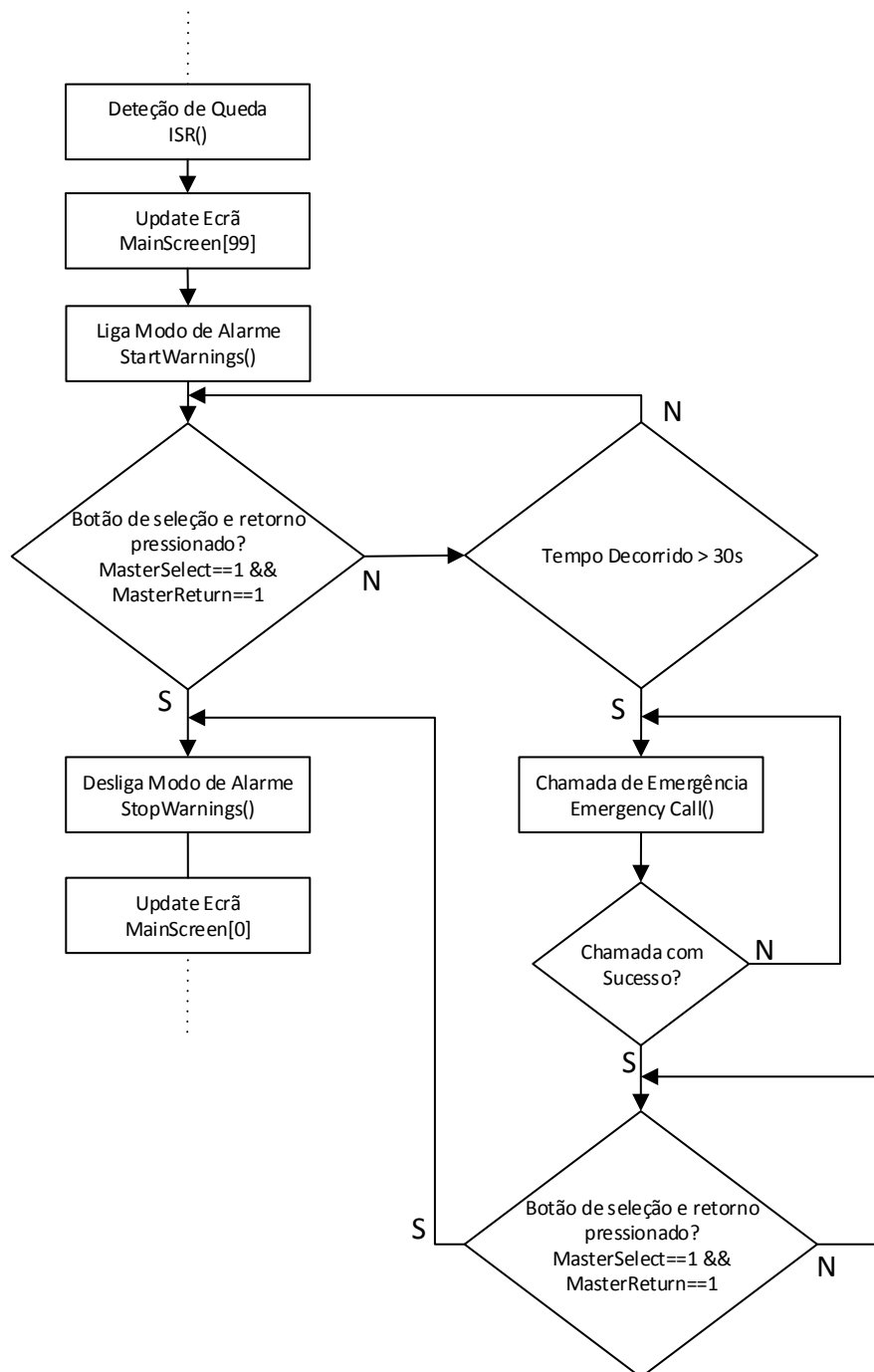


Figura 69 Fluxograma para aceder à aplicação de temperatura e humidade

7.5.4. MONITORIZAÇÃO DE SONO

A monitorização do sono é uma aplicação que permite gravar a informação recolhida pelos sensores presentes no equipamento. Para tal é necessário executar os seguintes passos:

1. Escolher o menu de monitorização de sono
2. Pressionar o botão de seleção
3. Escolher a opção de gravar para a memória interna ou enviar diretamente para o *smartphone*.

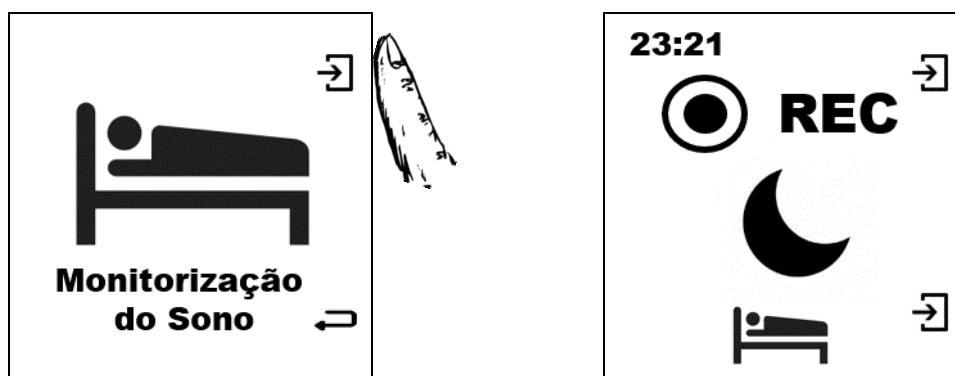


Figura 70 Menu de Monitorização do Sono

Na imagem da Figura 71 encontra-se o menu principal para entrar na aplicação de monitorização do sono.



Figura 71 Menu Principal para entrar na aplicação de Monitorização do Sono

No fluxograma que se segue é exemplificado a forma de funcionamento da aplicação de monitorização de sono. Após a seleção no menu principal, é necessário escolher se os

dados vão ser gravados para a memória, ou se a informação vai ser transmitida através de Bluetooth para um telemóvel, ou outro dispositivo compatível.

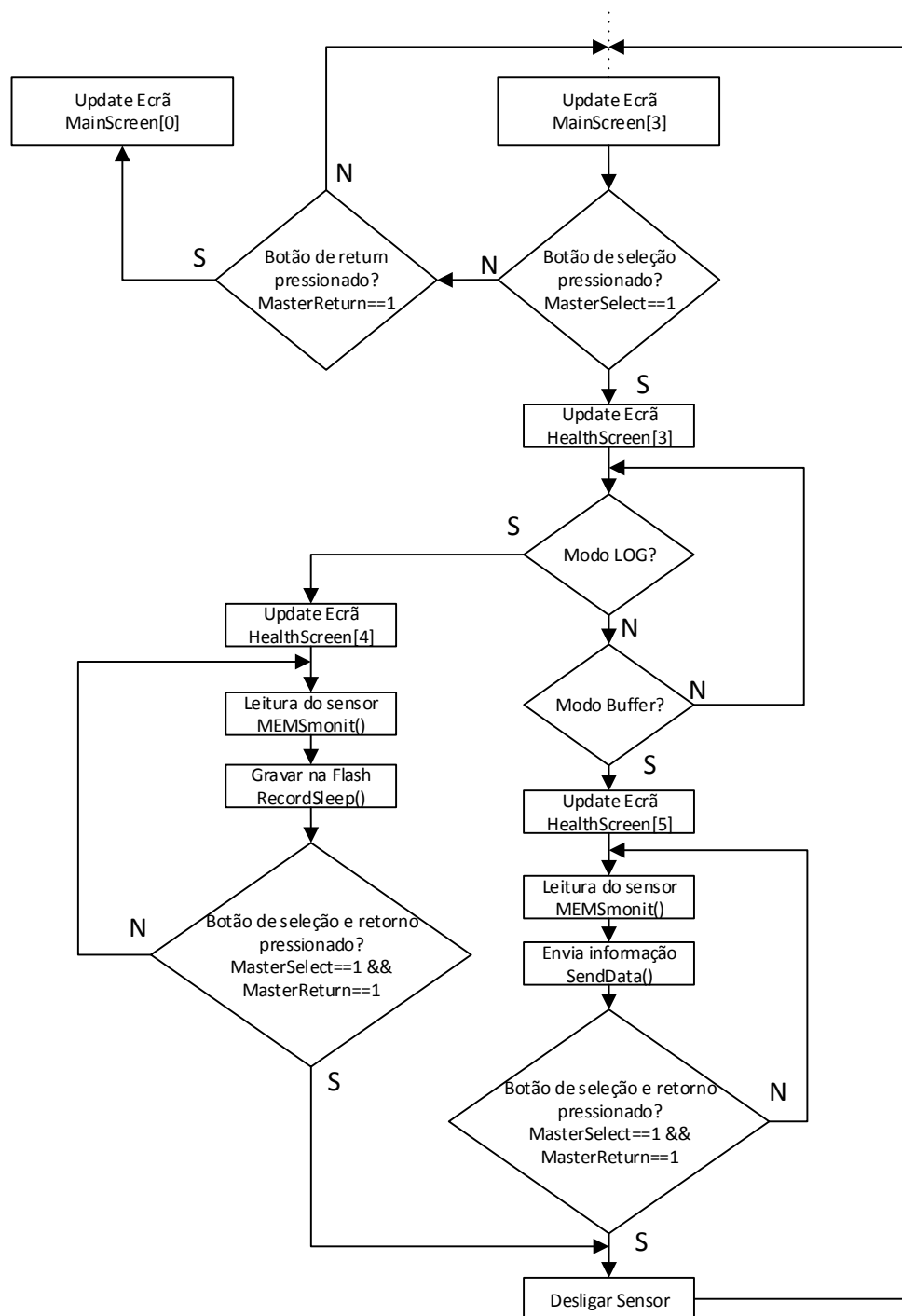


Figura 72 Fluxograma da aplicação de monitorização do sono

7.5.5. INFORMAÇÃO DE IDENTIFICAÇÃO

Para ativar o modo de identificação é necessário pressionar o botão de SOS por cinco segundos. Após isto o equipamento irá fazer um *broadcast* da informação contida, que pode ser lida por qualquer equipamento compatível.



Figura 73 Menu de Informação de Identificação

Quando o botão de SOS é pressionado por cinco segundos, no ecrã real aparece a imagem da Figura 74.



Figura 74 Aplicação em modo de transmissão de informação

No fluxograma que se segue é exemplificado o processo de iniciar a transmissão da informação guardada pelo dispositivo, para um equipamento compatível.

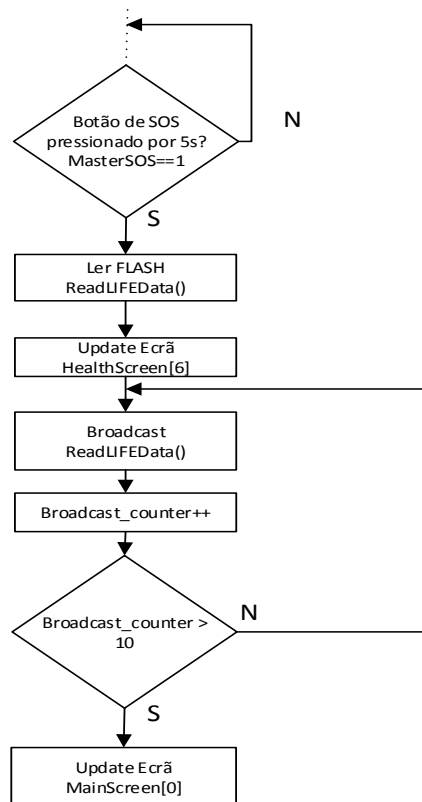


Figura 75 Fluxograma da aplicação de Identificação

7.5.6. CHAMADA DE EMERGÊNCIA

Em algumas situações o utilizador pode estar em uma situação em que se sinta mais débil, e sem capacidade para se cuidar. Nesta situação específica a deteção de queda pode não funcionar, porque a pessoa pode encontrar-se em posição não favorável à queda. Para tal existe o botão de emergência que pressionado por 10 segundos despoleta uma chamada para o número previamente definido através do *smartphone*.



Figura 76 Menu de indicação de chamada de emergência em curso

Na Figura 77 encontra-se a imagem que aparece quando o botão de SOS é pressionado por 10 segundos.



Figura 77 Indicação de chamada de emergência em curso no ecrã real

8. ARQUITETURA MECÂNICA

8.1. DISPOSIÇÃO DA ELETRÓNICA

Uma das principais preocupações da Arquitetura Mecânica é no agrupamento dos componentes eletrónicos, em que estes induzem algum atravancamento sendo necessário um estudo minucioso da sua disposição.

Os principais componentes do sistema em análise são:

- Ecrã (26,60 x 30,30 x 0,75 [mm])
- Bateria (15 x 28 x 4 [mm])
- Botão de emergência (15 x 5 x 0,50 [mm])
- Diversos dispositivos de medição

A disposição tanto interior como exterior dos componentes envolve uma análise que visa, não só a redução do incómodo dos equipamentos, como o *design* visual e acima de tudo a interação com o utilizador. Assim, a seleção do ecrã teve em conta (1) a variabilidade de utilizadores, que implica dimensões de pulso diferentes, (2) a quantidade de informação exposta e (3) a ocupação. Na Figura 78 é possível visualizar o aspeto do ecrã, e no Anexo A encontram-se os pormenores relativos às dimensões exteriores deste.



Figura 78 Ecrã Selecionado

A bateria exigiu um balanço que considerasse o seu período de duração, as suas características eletrónicas, a sua dimensão e custo. Este é o elemento que mais atravancamento induz no interior do equipamento, sendo necessária uma atenção redobrada no seu posicionamento. Este deve ter fácil acesso caso seja necessário a sua reposição, no entanto não tem impacto na componente do *design*.

O elemento exterior de mais importância é o botão de emergência, este permite ao utilizador que avise o serviço de Emergência vigente no país onde se encontra. Visto que faz parte do *design* particular do equipamento, optou-se por desenhar o botão para que este se adapta-se a todo o layout do equipamento, conforme a Figura 79.

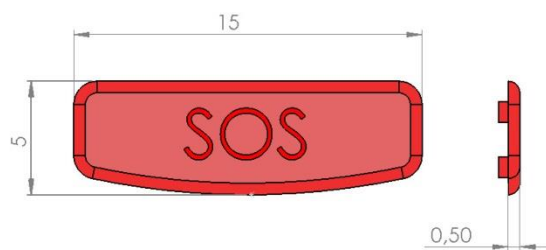


Figura 79 Design provisório do botão de emergência

Os restantes dispositivos de medição, como por exemplo do ritmo cardíaco e temperatura corporal, não afetam diretamente o *design* do equipamento, pois encontram-se na parte de trás deste. Esta disposição permite um contato direto dos sensores com o utilizador.

8.2. DESIGN PROTÓTIPO

Para o *design* inicial optou-se por uma abordagem adaptativa, que garanta a funcionalidade conjugada com um *design* atual.

Na parte frontal do equipamento existe apenas quatro interfaces Figura 80 (a), o botão de SOS, um LED, o ecrã e o slide, este último permite ao utilizador interagir a informação exposta no ecrã. Na lateral existem ainda mais dois botões, para a interação do utilizador com o software.



Figura 80 *Design* protótipo, parte frontal (a) e parte posterior (b).

Na parte posterior do dispositivo estão localizados os sensores que interagem com o utilizador, sendo de grande importância garantir o bom contato entre estes (sensores – utilizador). Assim sendo o *design* final tem que ter em conta a usabilidade do dispositivo por diferentes utilizadores, diferença essa que se reflete nas dimensões do pulso.

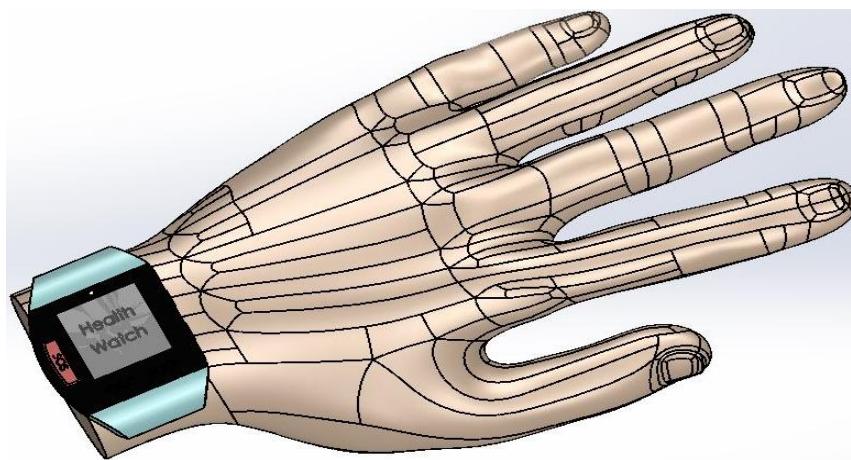


Figura 81 Simulação do aspeto do dispositivo quando em utilização

9. CONCLUSÕES

Os cenários de aplicação demonstrados, vão de encontro à tendência crescente da sociedade em monitorizar tudo ao seu redor, e em muitos casos a monitorização é fundamental para garantir a vida de pessoas. Estas aplicações médicas associadas à tecnologia *wearable* tornam o seu uso muito mais cómodo e prático.

Após a identificação dos requisitos e dos conceitos fundamentais aplicáveis, foi criada uma plataforma base capaz de interagir com diferentes tipos de hardware periférico. O trabalho efetuado, tal como a construção desta plataforma, foi essencial para a modularidade aplicacional, podendo futuramente adaptar a mesma a outro conjunto de aplicações que venham a ser identificadas. Para tal é necessário calibrar o sistema para a nova aplicação.

O hardware periférico abordado é específico para as aplicações de saúde, tais como batimento cardíaco, oxímetro, temperatura e humidade corporal, deteção de quedas, chapa de identificação e chamada de emergência, conforme o capítulo Descrição dos Componentes refere. Na fase de seleção de componente tentou-se escolher os de menor dimensão possível e de mais baixo consumo, uma vez que se trata de um equipamento com o objetivo de ser *wearable*, estas características foram sempre tidas em conta. Contudo, a maior parte destes componentes pode ter outras aplicações que não as descritas neste

trabalho, estando assim o trabalho facilitado para a adaptação a novos temas, bastando usar o hardware e o software específico.

O software também foi desenhado de maneira a ser o mais modular possível, de forma a poder adicionar e retirar aplicações. Para tal existe uma divisão muito clara das camadas criadas. A camada do sistema operativo em tempo real, que uniformiza qualquer componente de software. A camada de middleware, que gerência os recursos do sistema, os eventos e a interface gráfica. A de drivers, que foi desenhada para interagir diretamente com o hardware e filtrar algumas especificidades. Todo este pacote foi formatado especificamente para o cenário de aplicação mencionado.

O desenho referenciado no capítulo da Arquitetura mecânica, foi bem-sucedido para a fase de primeiro protótipo. O estudo do software de desenvolvimento mecânico foi realizado propositadamente para a execução deste projeto, mesmo sendo uma área fora do contexto do curso. Tentou-se adaptar ao máximo o formato de um relógio de pulso, às funcionalidades pretendidas, para manter sempre um formato o mais ergonómico possível. Contudo, futuramente é necessário estudar as forças mecânicas envolvidas, a forma de montagem em produção, e a ergonomia, de forma a construir um equipamento robusto, atrativo, e ao mesmo tempo capaz de se manter dentro do conceito de um dispositivo *wearable*.

Após este trabalho conclui-se que é viável avançar com a integração total de todo o sistema para as aplicações descritas no capítulo da Motivação e Enquadramento, de forma a maximizar as potencialidades de um sistema autónomo, ou somente como equipamento de monitorização independente.

Referências Documentais

- [1] Steve Mann, "Wearable computing as means for personal empowerment", ICWC-98, Fairfax VA, May 12-13 1998
- [2] Hendrik Witt, "Human-Computer Interfaces for Wearable Computers", A Systematic Approach to Development and Evaluation
- [3] T. T. Zhang; W. Ser; Goh Yam Thiam Daniel; Jianmin Zhang; Jufeng Yu; C. Chua; I. M. Louis, Sound Based Heart Rate Monitoring for Wearable Systems
- [4] S. Abbate, M. Avvenuti, P. Corsini, A. Vecchio, and J. Light, "Monitoring of human movements for fall detection and activities recognition in elderly care using wireless sensor network: a survey," 11 2010
- [5] MYO, <https://www.thalmic.com/en/myo/>
- [6] Sony SmartWatch 2, <http://www.sonymobile.com/pt/products/accessories/smartwatch-2-sw2/>
- [7] Jawbone Up, <https://jawbone.com/up>
- [8] MissFit Shine, <http://www.misfitwearables.com/>
- [9] Vinitha Khambadkar ; Eelke Folmer, GIST: a Gestural Interface for Remote Nonvisual Spatial Perception
- [10] EmBlocks IDE, <http://www.emblocks.org>
- [11] Codeblocks IDE, <http://www.codeblocks.org>
- [12] GCC - the GNU Compiler Collection, <http://gcc.gnu.org/>
- [13] SolidWorks Tutorials, <http://www.solidworks.com/sw/resources/solidworkstutorials.htm>
- [14] Apple iWatch, <http://www.t3.com/news/apple-iwatch-rumours-features-release-date>
- [15] Entrevista ao analista de equipamentos médicos Theo Ahadome, <http://wearablecomputingreview.com/tag/theo-ahadome/>
- [16] Organização Mundial de Saúde, <http://www.who.int/mediacentre/factsheets/fs344/en/>

Anexo A. Dimensões exteriores do ecrã

